

**UNIVERSIDAD POLITÉCNICA SALESIANA
SEDE QUITO**

CARRERA: INGENIERÍA DE SISTEMAS

**Trabajo de titulación previo a la obtención del título de:
INGENIEROS DE SISTEMAS**

**TEMA:
ANÁLISIS, DISEÑO E IMPLEMENTACIÓN DE UN SISTEMA PARA
INVENTARIO Y SEGURIDAD FÍSICA, UTILIZANDO ETIQUETAS RFID
(IDENTIFICACIÓN DE RADIO FRECUENCIA) PARA EL CONTROL DE
ACTIVOS INFORMÁTICOS EN LA EMPRESA SERTECPET S.A. DE
QUITO Y EL COCA.**

**AUTORES:
CARLOS ALBERTO ÁLVAREZ PÉREZ
JOSÉ RICARDO NAVAS AYALA**

**DIRECTOR:
JOSÉ LUIS AGUAYO MORALES**

Quito, mayo de 2015

DECLARATORIA DE RESPONSABILIDAD Y AUTORIZACIÓN DE USO DEL TRABAJO DE TITULACIÓN

Nosotros, autorizamos a la Universidad Politécnica Salesiana la publicación total o parcial de este trabajo de titulación y su reproducción sin fines de lucro.

Además, declaramos que los conceptos, análisis desarrollados y las conclusiones del presente trabajo son de exclusiva responsabilidad de los autores.

Quito, mayo de 2015

Carlos Alberto Álvarez Pérez
CC. 1722405279

José Ricardo Navas Ayala
CC. 0603885104

AGRADECIMIENTO

De manera especial agradecemos a nuestro director de tesis, Ing. José Luis Aguayo, quien con su esfuerzo, dedicación, conocimiento, experiencia, paciencia y motivación ha logrado en nosotros la culminación de este arduo camino con éxito.

Al finalizar el presente trabajo de tesis agradecemos a la Institución que nos abrió las puertas para emprender este proyecto y así alcanzar las metas y objetivos propuestos.

ÍNDICE

INTRODUCCIÓN	1
CAPÍTULO 1.....	2
PRESENTACIÓN Y FORMULACIÓN DEL PROBLEMA.....	2
1.1. Antecedentes	2
1.2. Planteamiento del problema	2
1.3. Formulación de objetivos	3
1.3.1. Objetivo general	3
1.3.2. Objetivos específicos	3
1.4. Alcance del problema	4
CAPÍTULO 2.....	7
ANÁLISIS Y REQUERIMIENTOS DEL PROYECTO	7
2.1 Situación actual	7
2.2 Requerimientos.....	8
2.3 Metodología de desarrollo.....	8
2.3.1 Extreme Programming (XP)	9
2.4 Herramientas de desarrollo.....	10
2.4.1 Recomendaciones en la creación de una base de datos.....	10
2.5 Levantamiento de información.....	11
CAPÍTULO 3.....	12
MARCO TEÓRICO	12
3.1. Marco conceptual	12
3.2. RFID: Identificación por radio frecuencia	13
3.2.1. Arquitectura.....	13
3.2.2. Tipos de tag	14
3.2.3. Composición de etiquetas RFID	16

3.2.4. Receptores:	18
3.3. Base de datos	18
3.3.1. Introducción a la base de datos SQL Server 2008	18
3.3.2. Conectar Java y la base de datos	20
3.4. Lenguaje de programación	21
3.4.1. Introducción al lenguaje de programación Java.....	21
3.4.2. Interfaz de persistencia Java (JPA) - entidades y managers	23
CAPÍTULO 4.....	30
DISEÑO E IMPLEMENTACIÓN	30
4.1. Diseño de la arquitectura del sistema	30
4.1.1. Diagrama de casos de uso	30
4.1.2. Documentación de casos de uso.....	31
4.2. Diseño de la base de datos.....	35
4.2.1. Descripción de tablas	35
4.3. Diseño del sistema.....	37
4.3.1. Creación de la persistencia en el sistema	37
4.3.2. Manejo de la información de la base de datos	46
4.3.3. Métodos de la clase EntityManager	52
4.3.4. Métodos de la clase EntityManagerFactory	54
4.3.5. Métodos de la clase EntityTransaction	55
4.3.6. Comunicación entre la aplicación y el lector RFID.....	56
4.4. Implementación del sistema	58
4.5. Administración y manejo del sistema	67
4.5.1. Pantalla inicio de sesión	67
4.5.2. Perfil administrador	67
4.5.3. Perfil visualizador	76
4.5.4. Perfil activos fijos	81

4.5.5. Perfil de seguridad.....	82
4.6. Instalación y ubicación de equipos.....	84
4.7. Análisis y resultados.....	87
CONCLUSIONES.....	90
RECOMENDACIONES.....	91
LISTA DE REFERENCIAS	92
ANEXOS	94

ÍNDICE DE FIGURAS

Figura 1. Croquis oficina matriz	4
Figura 2. Foto entrada oficina matriz.....	4
Figura 3. Croquis campamento base	5
Figura 4. Foto Entrada campamento base	5
Figura 5. Esquema general del funcionamiento de un sistema RFID	8
Figura 6. Esquema general de la metodología XP	9
Figura 7. Tag pasivo.....	15
Figura 8. Tag activo	15
Figura 9. Tag semipasivas.....	16
Figura 10. Principio maestro-esclavo aplicado al software, lector y la etiqueta.	18
Figura 11. Esquema de funcionamiento del controlador JDBC-ODBC.	21
Figura 12. Modelo de relación persistencia-Java.....	25
Figura 13. Diagrama de casos de uso.....	30
Figura 14. Diseño de la base de datos	35
Figura 15. Creación de persistencia	38
Figura 16. Inserción de la librería Toplink.....	38
Figura 17. Creación de conexión a la base de datos	39
Figura 18. Carga de driver para SQL	39
Figura 19. Configuración de parámetros SQL	40
Figura 20. Nivel de acceso SQL	40
Figura 21. Verificación de la conexión creada.....	41
Figura 22. Finalización de persistencia.....	41
Figura 23. Persistencia creada.....	42
Figura 24. Creación de entidad de persistencia.....	42
Figura 25. Selección de tablas en la entidad de persistencia.....	43
Figura 26. Selección de paquete.....	43
Figura 27. Finalización de entidades.....	44
Figura 28. Comprobación de clases de entidades	44
Figura 29. Creación de sentencias SQL	46
Figura 30. Creación de métodos set y get	47
Figura 31. Inserción de datos a la base	48
Figura 32. Método grabar.....	48

Figura 33. Actualización de datos a la base	49
Figura 34. Método modificar	50
Figura 35. Eliminación de datos a la base.....	51
Figura 36. Método eliminar	51
Figura 37. Liberia Giovynet.....	56
Figura 38. Clase puerto de comunicación	57
Figura 39. Clase conect	57
Figura 40. Clase read.....	58
Figura 41. Timer.....	58
Figura 42. Servidor HP Proliant DL380 G7	59
Figura 43. Computador HP Compaq 6000 Pro	59
Figura 44. Laptop Dell Latitude E6440	60
Figura 45. Lector RFID.....	60
Figura 46. Cable USB serial.....	61
Figura 47. Etiquetas RFID	61
Figura 48. Logo Netbeans	61
Figura 49. Logo SQL Server 2008 R2	62
Figura 50. Inicio de instalación SQL Server.....	62
Figura 51. Nueva instalación SQL Server.....	63
Figura 52. Clave de producto SQL Server 2008	63
Figura 53. Términos de licencia SQL Server 2008.....	64
Figura 54. Reglas auxiliares SQL Server 2008.....	64
Figura 55. Configuración servidor SQL Server 2008	65
Figura 56. Finalización de instalación SQL Server 2008	65
Figura 57. Comprobación instalación SQL Server 2008	66
Figura 58. Base de datos	66
Figura 59. Pantalla principal de inicio de sesión	67
Figura 60. Pantalla principal administrador	67
Figura 61. Pantalla ingreso de empleados perfil administrador	68
Figura 62. Pantalla de submenú empleados	68
Figura 63. Pantalla áreas de la empresa	69
Figura 64. Pantalla asignación entre usuarios	69
Figura 65. Pantalla ingreso de activos informáticos	70
Figura 66. Pantalla submenú activos.....	70

Figura 67. Pantalla submenú activos.....	71
Figura 68. Pantalla asignación de activos	71
Figura 69. Pantalla asignación de activos	72
Figura 70. Pantalla submenú perfiles	72
Figura 71. Pantalla creación de usuario	73
Figura 72. Pantalla submenú imprimir.....	73
Figura 73. Pantalla impresión de reporte	74
Figura 74. Diagrama de procesos perfil administrador.....	75
Figura 75. Pantalla principal perfil visualizador.....	76
Figura 76. Pantalla ingreso de empleados perfil visualizador.....	76
Figura 77. Pantalla áreas de la empresa perfil visualizador	77
Figura 78. Pantalla submenú asignación entre empleados perfil visualizador.....	77
Figura 79. Pantalla de activos informáticos perfil visualizador	78
Figura 80. Pantalla submenú tipos de activos informáticos perfil visualizador.....	78
Figura 81. Pantalla asignación de activos informáticos perfil visualizador.....	79
Figura 82. Pantalla de configuración perfil visualizador	79
Figura 83. Pantalla impresión de reporte perfil visualizador	80
Figura 84. Diagrama de procesos perfil visualizador.....	80
Figura 85. Pantalla principal perfil activos fijos	81
Figura 86. Pantalla activos informáticos perfil activos fijos.....	81
Figura 87. Pantalla submenú tipos de activos informáticos perfil activos fijos.....	82
Figura 88. Diagrama de procesos perfil activos fijos.....	82
Figura 89. Pantalla principal perfil de seguridad	83
Figura 90. Pantalla principal perfil de seguridad	83
Figura 91. Diagrama de procesos perfil activos fijos.....	84
Figura 92. Guardia garita Eloy Alfaro	85
Figura 93. Guardia garita Arroyo del Rio	85
Figura 94. Guardia garita campamento base.....	86
Figura 95. Administrador del sistema	86

ÍNDICE DE TABLAS

Tabla 1. Memoria de los tag.....	17
Tabla 2. Documentación de casos de uso.....	31
Tabla 3. Comparación entre costo de la consultora y el sistema	88

ÍNDICE DE ANEXOS

Anexo 1. Guía de entrevista realizada al coordinador de activos fijos	94
Anexo 2. Guía de entrevista realizada al coordinador de TI.....	96
Anexo 3. Guía de entrevista realizada al supervisor de seguridad.....	98
Anexo 4. Levantamiento de Información.....	99
Anexo 5. Guía de entrevista realizada al coordinador de activos fijos	140
Anexo 6. Guía de entrevista realizada al coordinador de TI.....	141
Anexo 7. Guía de entrevista realizada al supervisor de seguridad.....	142
Anexo 8. Factura de consultora para la realización de los informes de activos.....	143

RESUMEN

Sertecpet SA posee un gran número de activos electrónicos desplegados en sus diferentes locaciones y por este motivo se desconoce su ubicación real, la cantidad total de equipos y sus características, ocasionando miles de dólares en pérdidas a la compañía debido a que se realizan compras innecesarias de equipos para los diferentes proyectos, desconociendo si existen equipos disponibles para su reutilización.

El presente proyecto pretende efectuar un eficiente control de inventario de activos electrónicos y seguridad física a través de un sistema programado el cual permite acceder al mismo mediante perfiles de usuario, el sistema será controlado mediante sensores en los diferentes accesos de la matriz y el Campo Base, alimentado mediante etiquetas RFID ubicadas en los activos electrónicos previamente instruidos con la información específica de los mismos, los datos serán almacenados en una base de datos SQL Server 2008, se presentaran reportes los cuales están diseñados bajo la herramienta Ireport.

Los resultados del sistema implementado muestra que los reportes se los pueden obtener en un 0.27% del tiempo que antes requería, al 1% del costo que antes se pagaba, mostrando así que los resultados del sistema satisfacen los inconvenientes que poseía la empresa.

ABSTRACT

Sertecpet SA has a large number of electronic assets deployed in different locations and for this reason its actual location is unknown, the total amount of equipment and their characteristics, causing thousands of dollars in losses to the company because unnecessary purchases are made equipment for different projects, ignoring whether there equipment available for reuse.

This project aims to make efficient inventory control Active electronic and physical security through a programmed system which enables access it using user profiles, the system will be controlled by sensors at different entrances of the matrix and the Base Camp, powered by RFID tags located in previously instructed electronic assets with specific information from them, the data will be stored in a SQL Server 2008 data reports which are designed under the Ireport tool is presented.

The results of the implemented system shows that the reports are obtainable in 0.27% of the time previously required, 1% of the cost was paid before, showing that the results meet the drawbacks of the system held by the company.

INTRODUCCIÓN

En el capítulo uno se detalla la Presentación y Formulación del Problema, en el cual se establece la problemática que posee la empresa así como el objetivo general, los objetivos específicos y el alcance que tendrá el presente proyecto.

En el capítulo dos se determina el Análisis y Requerimientos del Proyecto, se explica la situación actual y los requerimientos, en base a estos se ha determinado las herramientas y la metodología que será utilizada para el desarrollo del proyecto.

En el capítulo tres Marco Teórico involucra los conceptos principales que se utilizan a lo largo del proyecto.

En el capítulo cuatro se establece el Diseño e Implementación del Sistema, en el cual se presenta el desarrollo y la implementación de las partes más relevantes del sistema. Además se exponen las pruebas y resultados en los cuales se puede evidenciar que los objetivos establecidos fueron cumplidos y las necesidades previamente determinadas de la empresa satisfechas.

En Conclusiones y Recomendaciones damos nuestras opiniones acerca del desarrollo e implementación del sistema, así como nuestras experiencias obtenidas a lo largo del proyecto y las mejoras a considerarse en el futuro.

En el capítulo Anexos se adjunta toda la información relevante que ha sido importante para la consecución del proyecto.

CAPÍTULO 1

PRESENTACIÓN Y FORMULACIÓN DEL PROBLEMA

1.1. Antecedentes

Sertecpet S.A. es una empresa que brinda servicios petroleros en Quito y en la región amazónica, por lo que posee numerosas locaciones como pozos petroleros, bodegas y campamentos, con lo que el ingreso y salida de equipos informáticos es muy frecuente de sus instalaciones y actualmente no posee ningún sistema para llevar un control eficiente y seguro de ellos.

La empresa Sertecpet S.A., en su afán por mejorar el control de los activos informáticos tanto de sus usuarios internos y externos ha considerado implementar el sistema de etiquetas RFID (Identificación por Radio Frecuencia), sustituyendo a las etiquetas adhesivas con código de identificación simple que hoy en día posee.

Este proyecto busca mejorar el control de activos en la empresa ayudando a que todos los dispositivos electrónicos, estén siendo monitoreados, para evitar la pérdida y mal uso de los mismos a través del sistema de localización e inventario.

La implementación traerá importantes beneficios, ya que la adquisición de equipos será menor, se obtendrá información diaria de los mismos en cualquier parte de Sertecpet S.A. y se tendrá un mayor control y seguridad en el ingreso y salida de los activos informáticos de la empresa.

1.2. Planteamiento del problema

Sertecpet S.A. posee un gran número de activos electrónicos desplegados en sus diferentes locaciones y por este motivo se desconoce su ubicación real, la cantidad total de equipos y sus características, ocasionando miles de dólares en pérdidas a la compañía debido a que se realizan compras innecesarias de equipos para los diferentes proyectos, desconociendo si existen equipos disponibles para su reutilización.

Esto ocasiona que haya un desajuste en el presupuesto establecido para el área de TI lo que lleva a paralizar otros proyectos que se tienen planificados.

Otro de los inconvenientes es al momento de la salida de equipos ya que el control en los accesos es deficiente por lo que cualquier persona puede sustraerse los activos de la compañía, por este motivo es necesario implementar un sistema eficiente de control de inventario de activos electrónicos y seguridad física (RFID).

Esta problemática involucra a las áreas de TI, Finanzas, Control de Activos Fijos, Seguridad Física, por tal motivo se trabajará de manera conjunta, con las áreas antes mencionadas, para llevar a cabo este proyecto.

1.3. Formulación de objetivos

1.3.1. Objetivo general

Analizar, diseñar e implementar un sistema de inventario y seguridad física utilizando etiquetas RFID (Identificación de radio frecuencia), el cual permita tener un registro, control y seguridad de los activos informáticos de la empresa Sertecpet S.A de Quito y el Coca.

1.3.2. Objetivos específicos

- Programar un software el cual permita inventariar, emitir reportes y alertar el movimiento de los equipos electrónicos, almacenados a través de una base de datos.
- Configurar las etiquetas (Tags) para almacenar la información con los datos específicos de cada activo mediante los dispositivos adecuados.
- Ensamblar e instruir los dispositivos electrónicos de lectura (Sensores) de las etiquetas RFID para que recepten la información previamente almacenada en estas.
- Presentar el producto final a la empresa Sertecpet S.A. y a la Universidad Politécnica Salesiana.

1.4. Alcance del problema

El presente proyecto será implementado en la oficina matriz en Quito cuya dirección es: Eloy Alfaro N37-25 y Jose Correa (ver Figura 1):

Croquis oficina matriz



Figura 1. Croquis oficina matriz

Fuente: Google Maps

Acceso a la Entrada Principal a las Oficinas de la Matriz (ver Figura 2)

Entrada oficina matriz

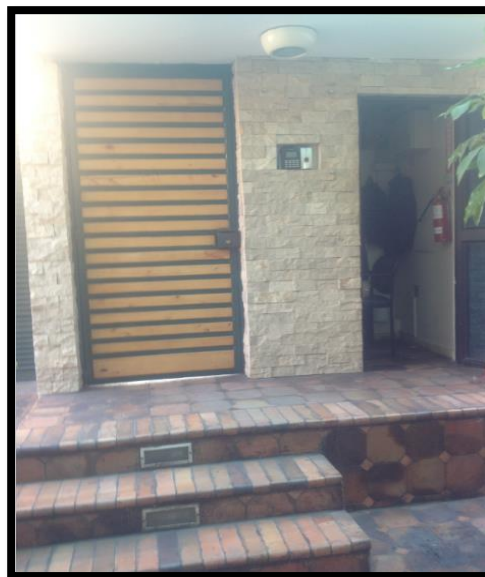


Figura 2. Foto entrada oficina matriz

Elaborado por: J. Navas y C. Álvarez

Y en el Campamento Base Carlos López Robayo ubicado en la Provincia de Orellana en el Km 5 vía al Coca (ver Figura 3).

Croquis campamento base



Figura 3. Croquis campamento base
Fuente: Google Earth

Acceso a la Entrada Principal al Campamento Base Carlos López Robayo (ver Figura 4):

Campamento Base



Figura 4. Foto Entrada campamento base
Elaborado por: J. Navas y C. Álvarez

Se considera activo fijo, según las políticas de la compañía, todo bien registrado con un costo mayor o igual a \$700.

El proyecto tomará en cuenta como activos fijos informáticos, los siguientes dispositivos tangibles:

- Laptops.
- Computadores Desktop.
- Impresoras.
- Servidores.
- Firewall.
- Switch.
- Access Point (AP).

Seguridad Física es el nombre por el cual se conoce al área de guardianía en la empresa Sertecpet S.A.

El sistema de control de acceso permitirá normalmente la entrada y salida de los activos en horarios de oficina que son de 8:00 a 17:30 de Lunes a Viernes y si el caso amerita en Feriados y fines de semana previa petición documentada de gerencia, cualquier situación que se presente fuera de estos horarios con el ingreso o salida de activos estos serán retenidos en guardianía.

CAPÍTULO 2

ANÁLISIS Y REQUERIMIENTOS DEL PROYECTO

2.1 Situación actual

Según la entrevista realizada al responsable de Activos fijos (la guía de la entrevista consta en el Anexo 1) Sertecpet S.A. lleva un control de los activos electrónicos con etiquetas adhesivas en las que va impreso un número de activo, estas etiquetas son asignadas en el momento que el equipo llega al departamento de IT.

Mediante el número de activo que está en la etiqueta se genera un acta de entrega impresa la misma que lleva los datos básicos de la persona a ser entregada, este proceso se lo realiza manualmente, ocasionando pérdida de tiempo al generar este documento. Las etiquetas no poseen ningún sistema de seguridad que impida la salida de los activos sin consentimiento de las áreas encargadas del control de la seguridad física en la empresa.

Para la salida y entrada de activos se realiza una revisión ocasional del equipaje que lleva el personal externo o interno por la guardianía de seguridad (Seguridad Física) en los accesos (la guía de entrevista utilizada consta en el Anexo 2), en el caso de que un equipo sea sacado de las instalaciones un elemento de guardianía anota el número de activo y la marca del equipo en una hoja llenada a mano.

Esto ocasiona que no se tenga un control eficiente de los equipos que salen de las instalaciones ya que el rastreo no es permanente porque no se respeta los lineamientos impuestos por la guardianía al restar tiempo operativo para sus actividades laborales. El proceso carece de registros actualizados, historial inexistente de los activos y desperdicio de suministros de oficina al generar el registro de dichos activos.

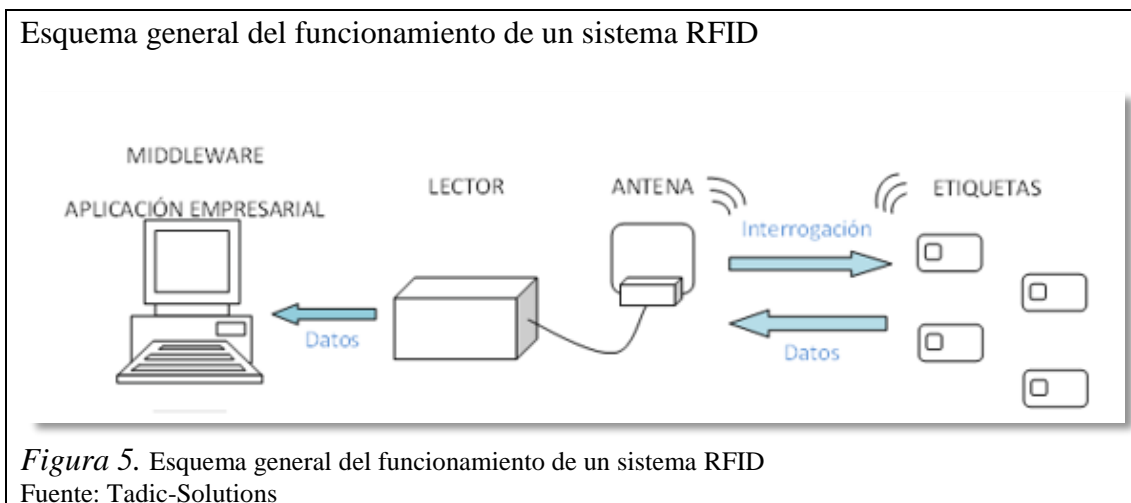
Luego de entrevistar al Coordinador de TI (la guía de la entrevista consta en el Anexo 3) se determinó la necesidad de crear un sistema de control automático de activos electrónicos con etiquetas RFID.

2.2 Requerimientos

Para el presente proyecto según entrevistas (Anexo 1, Anexo 2, Anexo 3) se necesitarán los siguientes componentes para la puesta en marcha del sistema.

Según Tadic Solutions (2014) los componentes básicos de un sistema RFID son: Tag, lector, antena RF y sistema gestor de información (ver Figura 5). Un sistema RFID no está completo si carece de alguno de estos cuatro elementos.

El modo de operación de un sistema RFID básico consiste en la identificación localizada y automática de objetos etiquetados. Dentro de este objetivo final, cada uno de los componentes del sistema tiene su función particular que permite llevar a cabo el proceso de identificación.



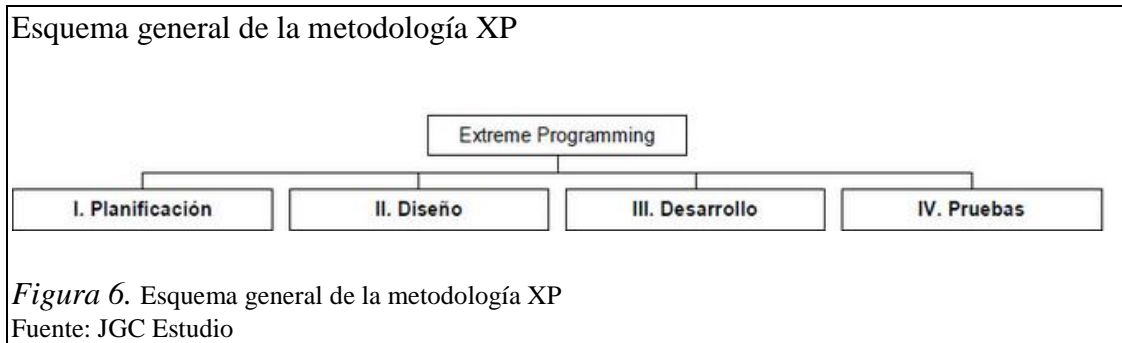
Adicional a esto se utilizará un servidor para la base de datos, que almacene los datos emitidos por las etiquetas y receptados por las antenas. El aplicativo tendrá una conexión tipo cliente-servidor (ODBC) a la base de datos.

2.3 Metodología de desarrollo

Se utilizará la Metodología de Programación Extrema (XP), debido a sus características que se adaptan al proyecto propuesto.

2.3.1 Extreme Programming (XP)

La metodología para Somerville (2005) consiste en una programación rápida o extrema, cuya particularidad es tener como parte del equipo, al usuario final, pues es uno de los requisitos para llegar al éxito del proyecto (Figura 6).



a. Bases de la metodología XP

Esta metodología, según Somerville (2005), está basada entre las más importantes:

- **Pruebas unitarias:** se basa en las pruebas realizadas a los principales procesos, de tal manera que adelantándonos en algo hacia el futuro, se puedan hacer pruebas de las fallas que pudieran ocurrir.
- **Re fabricación:** se basa en la reutilización de código, para lo cual se crean patrones o modelos estándares, siendo más flexible al cambio.
- **Programación en pares:** una particularidad de esta metodología es que propone la programación en pares, la cual consiste en que dos desarrolladores participen en un proyecto en una misma estación de trabajo. Cada miembro lleva a cabo la acción que el otro no está haciendo en ese momento. (pág. 85)

Dentro de esta metodología existen las siguientes proposiciones:

- Empieza en pequeño y añade funcionalidad con retroalimentación continua
- El manejo del cambio se convierte en parte sustantiva del proceso
- El costo del cambio no depende de la fase o etapa

- No introduce funcionalidades antes que sean necesarias
- El cliente o el usuario se convierten en miembro del equipo

Según Somerville (2005) Indica que lo fundamental en este tipo de metodología es:

- La comunicación, entre los usuarios y los desarrolladores
- La simplicidad, al desarrollar y codificar los módulos del sistema
- La retroalimentación, concreta y frecuente del equipo de desarrollo, el cliente y los usuarios finales (pág. 88)

2.4 Herramientas de desarrollo

Como sugiere Universidad Técnica de Machala (2013) se ha decidido utilizar Netbeans de Java ya que: “permite a los desarrolladores crear y depurar aplicaciones web y móviles utilizando HTML5, JavaScript, y las normas CSS3. Otros puntos destacados incluyen mejoras continuas del IDE para Groovy, PHP, JavaFX”.

2.4.1 Recomendaciones en la creación de una base de datos

Según Silberschatz (2002) los sistemas de bases de datos se diseñan para gestionar grandes cantidades de información. La gestión de los datos implica tanto la definición de estructuras para almacenar la información como la provisión de mecanismos para la manipulación de la información. Además, los sistemas de bases de datos deben proporcionar la fiabilidad de la información almacenada, a pesar de las caídas del sistema o los intentos de acceso sin autorización. Si los datos van a ser compartidos entre diversos usuarios, el sistema debe evitar posibles resultados anómalos.

Un sistema de bases de datos es una colección de archivos interrelacionados y un conjunto de programas que permitan a los usuarios acceder y modificar estos archivos. Uno de los propósitos principales de un sistema de bases de datos es proporcionar a los

usuarios una visión abstracta de los datos. Es decir, el sistema esconde ciertos detalles de cómo se almacenan y mantienen los datos.

Existen básicamente 2 arquitecturas de bases de datos: Modelo aislado o Standalone y Modelo Cliente-Servidor (pág. 145).

a. Modelo Standalone

En este modelo, la base de datos y el servidor residen en la misma máquina, por lo que contienen el mismo sistema de archivos. Solo puede acceder un usuario concurrentemente y no está en red.

b. Modelo cliente-servidor

Para Silberschatz (2002) el cliente y el servidor pueden estar en computadores diferentes, pueden acceder varios usuarios concurrentemente. Esta arquitectura es apropiada para colocarla en la red, lo que permite el acceso rápido a la base de datos donde y cuando se quiera. La seguridad de la base de datos está representada por una clave de administrador y un nombre de usuario. (pág. 148)

Actualmente la empresa Sertecpet S.A. posee licencias de SQL Server 2008, según indica el Ing. Daniel Vallejo (Administrador de Inteligencia Informática), y ha mostrado ser estable para el procesamiento transaccional.

2.5 Levantamiento de información

Mediante un registro manual se procedió a tomar los datos de todos los activos informáticos de la compañía tanto en Quito y el Coca utilizando la ficha de Informe de Equipos Informáticos (Anexo 4).

CAPÍTULO 3

MARCO TEÓRICO

3.1.Marco conceptual

Para NX-ID (2011), la Tecnología RFID - “**Radio Frequency Identification**” (identificación por radiofrecuencia) es la identificación automática que almacena y recupera datos de forma remota a través de una etiqueta RFID (Tag); con RFID es posible realizar lecturas simultáneas de objetos, productos, vehículos o personas, agilizando sustancialmente los procesos de identificación; es posible leer los datos de los Tags integrados en los objetos o productos, aun cuando no existe línea visual entre el producto y el lector de RFID.

Los sistemas RFID constan de tres componentes básicos: una antena, una Lectora y un Tag (etiqueta RF). El funcionamiento de RFID se basa en este sencillo concepto: Se envía una señal a un dispositivo eléctrico, que se activa y devuelve la señal (sistema pasivo) o emite una señal (sistema activo).

La ventaja más importante de la Tecnología RFID, es la capacidad de supervisión electrónica de los productos. A diferencia del sistema de códigos de barras, el inventario puede supervisarse en todos y cada uno de los puntos de la cadena de suministros; erradica las desventajas de los códigos de barras, que consisten en que cada artículo debe escanearse manualmente, sólo pueden almacenar una cantidad limitada de información y que deben cambiarse cada vez que cambia la información. Los códigos de barras sucios o dañados resultan ilegibles.

Otra de las Ventajas de RFID es que no requieren del contacto directo ni línea de visión para funcionar correctamente. Las etiquetas pueden leerse en todo tipo de entornos, ya sea con nieve, niebla, hielo, pintura, suciedad u otras condiciones. Las etiquetas RFID también pueden leerse muy rápido: en la mayoría de los casos responden en menos de 100 milisegundos.

3.2.RFID: Identificación por radio frecuencia

RFID utiliza la radiofrecuencia para el almacenamiento y recuperación de datos remoto. El sistema en general se compone de un lector, un TAG y un sistema de procesamiento de datos.

Para Dargam (2004) el propósito fundamental de la tecnología RFID es transmitir la identidad de un objeto (similar a un número de serie único) mediante ondas de radio. Las tecnologías RFID se agrupan dentro de las denominadas Auto ID (automatic identification, o identificación automática).

Según Tanembaum (2011) a menudo, las etiquetas se parecen a pegatinas que se pueden colocar, por ejemplo, en pares de pantalones, en los estantes de una tienda. La mayor parte de la etiqueta está ocupada por una antena que es impresa en la misma. Un pequeño punto en el centro es el circuito integrado de RFID.

Alternativamente, las etiquetas RFID se pueden integrar en un objeto, tal como una licencia de conducir, carnet, etc. En ambos casos, las etiquetas no tienen una batería y deben reunir la energía de los radios transmisiones de un lector RFID cercano a ser leído. Este tipo de etiqueta se llama Tag Pasiva para distinguirlas de las etiquetas que tienen baterías Tag Activas (pág. 90).

3.2.1. Arquitectura

Según Texas Instruments (2012), “el modo de funcionamiento de los sistemas RFID es simple. La etiqueta RFID, que contiene los datos de identificación del objeto al que se encuentra adherido, genera una señal de radiofrecuencia con dichos datos. Esta señal puede ser captada por un lector RFID, el cual se encarga de leer la información y pasarla en formato digital a la aplicación específica que utiliza RFID”.

Para Pandaid (2012) un sistema RFID consta de los siguientes tres componentes:

- a. **Etiqueta RFID o transpondedor:** compuesta por una antena, un transductor radio y un material encapsulado o chip. El propósito de la antena es permitirle al chip, el cual contiene la información, transmitir la información de

identificación de la etiqueta. Existen varios tipos de etiquetas. El chip posee una memoria interna con una capacidad que depende del modelo y varía de una decena a millares de bytes.

Existen varios tipos de memoria:

- **Solo lectura:** el código de identificación que contiene es único y es personalizado durante la fabricación de la etiqueta.
 - **De lectura y escritura:** la información de identificación puede ser modificada por el lector.
 - **Anticolisión:** se trata de etiquetas especiales que permiten que un lector identifique varias al mismo tiempo (habitualmente las etiquetas deben entrar una a una en la zona de cobertura del lector).
- b. **Lector de RFID o transceptor:** compuesto por una antena, un transceptor y un decodificador. El lector envía periódicamente señales para ver si hay alguna etiqueta en sus inmediaciones. Cuando capta una señal de una etiqueta (la cual contiene la información de identificación de esta), extrae la información y se la pasa al subsistema de procesamiento de datos.
- c. **Subsistema de procesamiento de datos o middleware RFID:** proporciona los medios de proceso y almacenamiento de datos.

3.2.2. Tipos de tag

i. Tag pasivo

Son Etiquetas RFID que no poseen alimentación, se utiliza la energía del interrogador (lector) tanto para recibir como para transmitir (ver Figura 7). Se utilizan técnicas de construcción de circuitos integrados para su fabricación, alcanzando tamaños pequeños y llegando a estar ubicados en diferentes superficies como vidrios de automóviles.

El método de fabricación hace que sean económicos y se puedan fabricar en cadena. Al no contar con batería, están diseñados para tener poco alcance (distancias de lectura de 10 cm a unos pocos metros), no se comportan muy

bien en ambientes ruidosos (paredes metálicas, ruido electromagnético y todas las superficies que reflejen la señal de radiofrecuencia).

Tag pasivo



Figura 7. Tag pasivo
Fuente: Dipolerfid

ii. Tag activo

A diferencia de los Tags pasivos, poseen fuente de alimentación, por lo tanto generan menos errores de lectura. Tiene alcances de cientos de metros y se pueden utilizar para monitorear diversas variables (temperatura, posición). Debido a su fuente de alimentación son más costosos, tienen menor vida útil y son más grandes que los Tags pasivos (Ver Figura 8).

Tag activo

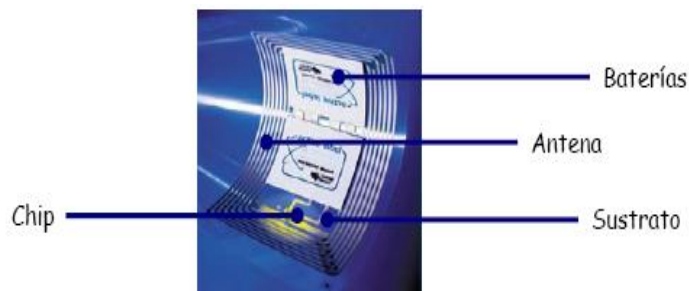


Figura 8. Tag activo
Fuente: Dipolerfid

iii. Tag semipasivas

Las etiquetas semipasivas se parecen a las activas en que poseen una fuente de alimentación propia, aunque en este caso se utiliza principalmente para alimentar el microchip y no para transmitir una señal. La energía contenida en la radiofrecuencia se refleja hacia el lector como en una etiqueta pasiva. Un uso alternativo para la batería es almacenar información propagada desde el lector

para emitir una respuesta en el futuro. Las etiquetas sin batería deben responder reflejando energía de la portadora del lector al vuelo (ver Figura 9).

Tag semipasivas



Figura 9. Tag semipasivas

Fuente: Stardustxs

La batería puede permitir al circuito integrado de la etiqueta estar constantemente alimentado y eliminar la necesidad de diseñar una antena para recoger potencia de una señal entrante. Por ello, las antenas pueden ser optimizadas para utilizar métodos de backscattering (*reflexión de las ondas, partículas, o las señales de vuelta a la dirección de la que proceden*). Las etiquetas RFID semipasivas responden más rápidamente, por lo que son más fuertes en el ratio de lectura que las pasivas.

Este tipo de etiqueta tiene una fiabilidad comparable a la de las activas, a la vez que pueden mantener el rango operativo de una pasiva. También suelen durar más tiempo que las activas.

3.2.3. Composición de etiquetas RFID

El Tag (etiqueta) consta de dos componentes fundamentales:

- Antena: a mayor dimensión del Tag, se tendrá que aumentar el tamaño de su antena, obteniendo más alcance.
- Memoria: oscila entre 128 y 512 bits. Está estructurada en 4 bancos.

3.2.3.1. Cómo funcionan los tag

- Usa el típico protocolo half-duplex.
- El Reader envía una señal de radiofrecuencia para preguntar información de los tags que se encuentren en el radio de cobertura de dicho reader.
- El Tag recibe dicha señal, y la utiliza como señal de alimentación. Dicha señal es a su vez modulada en amplitud por la información, de tal manera que el Tag termina reflejando la señal de entrada inicial modulada ASK (de respuesta).
- El receiver a su vez adquiere la señal reflejada y la demodula, obteniendo así la información del Tag que ha contestado.

3.2.3.2. Memoria de los tags

Tabla 1. Memoria de los tag

Banco 11	Usuario
Banco 10	Tid
Banco 01	Epc
Banco 00	Reservado

Nota: Estructura estándar que utiliza una memoria tag.

Elaborado por: J. Navas y C. Álvarez.

- **Usuario:** no es obligatorio, pero esta zona de memoria puede albergar información vital en la optimización de procesos.
- **TID:** identificador para el Tag en sí mismo (un n° de serie a parte del EPC).
- **EPC:** el campo EPC contiene el n° EPC del Tag en cuestión (Cada objeto con tecnología RFID lleva asignado un número individual y único, llamado EPC).
- **Reservado:** contiene las contraseñas para deshabilitar el Tag (Kill Tag) como las contraseñas de acceso que contiene dicho protocolo.

El EPC se puede utilizar para identificar:

- Productos de cualquier tipo.
- Personal, pacientes e instrumentos médicos.

- Identificación activos.
- Identificación de flota vehicular.

3.2.4. Receptores

También llamado interrogador, es uno de los componentes principales del sistema RFID, ya que es el encargado de realizar la comunicación con las etiquetas y la aplicación (software) que generalmente se encuentra en un computador. Lo anterior sucede sólo cuando las etiquetas se encuentran dentro de la zona de interrogación, la que consiste en el área en la que el lector lee/escribe datos desde o hacia la etiqueta.

La relación que existe entre el lector y la etiqueta es de maestro-esclavo, donde el lector actúa como maestro y las etiquetas como esclavos, al igual que entre la aplicación y el lector (maestro-esclavo), donde la aplicación actúa como maestro y el lector como esclavo, como se observa en la Figura 10.

Principio maestro-esclavo aplicado al software, lector y la etiqueta

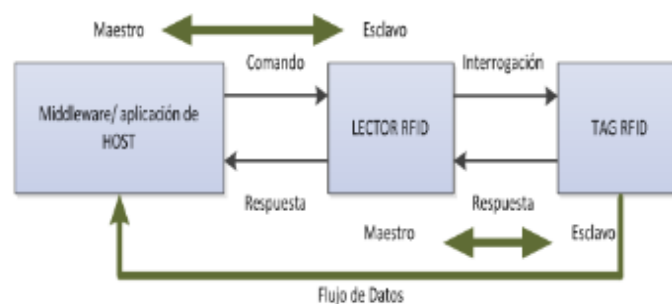


Figura 10. Principio maestro-esclavo aplicado al software, lector y la etiqueta.

Fuente: RFID Readers

3.3.Base de datos

3.3.1. Introducción a la base de datos SQL Server 2008

“Es un sistema para la gestión de bases de datos producido por Microsoft basado en el modelo relacional. Sus lenguajes para consultas son T-SQL y ANSI SQL. Microsoft SQL Server constituye la alternativa de Microsoft a

otros potentes sistemas gestores de bases de datos como son Oracle, PostgreSQL o MySQL” (Microsoft, 2014).

Según Vásquez (2012) SQL Server 2008 incluye una gran cantidad de características que permiten una gestión más racional y eficaz del mismo, aumentan el rendimiento, la escalabilidad y la estabilidad del servidor y, permiten una configuración avanzada a nivel de servicios, seguridad del servidor, etc.

Para Vásquez (2012) todas estas características se agrupan dentro de SQL Server 2008 se describen a continuación:

- Soporte de transacciones.
- Soporta procedimientos almacenados.
- Incluye también un entorno gráfico de administración, que permite el uso de comandos DDL y DML gráficamente.
- Permite trabajar en modo cliente-servidor, donde la información y datos se alojan en el servidor y los terminales o clientes de la red sólo acceden a la información.
- Además permite administrar información de otros servidores de datos.

Este sistema incluye una versión reducida, llamada MSDE con el mismo motor de base de datos pero orientado a proyectos más pequeños, que en sus versiones 2005 y 2008 pasa a ser el SQL Express Edition, que se distribuye en forma gratuita.

Es común desarrollar completos proyectos complementando Microsoft SQL Server y Microsoft Access a través de los llamados **ADP** (Access Data Project). De esta forma se completa la base de datos (Microsoft SQL Server), con el entorno de desarrollo (VBA Access), a través de la implementación de aplicaciones de dos capas mediante el uso de formularios Windows.

En el manejo de SQL mediante líneas de comando se utiliza el SQLCMD, osql, o PowerShell.

Para el desarrollo de aplicaciones más complejas (tres o más capas), *Microsoft SQL Server* incluye interfaces de acceso para varias plataformas de desarrollo, entre ellas .NET, pero el servidor sólo está disponible para Sistemas Operativos

3.3.2. Conectar Java y la base de datos

Según Arias la capacidad para acceder a bases de datos desde Java la ofrece la API JDBC (*Java DataBase Connectivity*). JDBC es un estándar para manejar bases de datos en Java.

ODBC (*Open Database Connectivity*) es un estándar de *Windows* para manejar bases de datos, de forma que cualquier programa en *Windows* que desee acceder a bases de datos genéricas debe usar este estándar (Arias, 2012).

i. Controlador JDBC-ODBC

Se establece un puente entre JDBC y ODBC. Este controlador convierte todas las llamadas JDBC a llamadas ODBC y realiza la conversión correspondiente de los resultados.

Este controlador es el que implementa la funcionalidad de todas las clases de acceso a datos y proporciona la comunicación entre el API JDBC y la base de datos real.

La necesidad del JDBC, a pesar de la existencia de ODBC, viene dada porque ODBC es un interfaz escrito en lenguaje C, que al no ser un lenguaje portable, haría que las aplicaciones Java también perdiesen la portabilidad.

Esquema de funcionamiento del controlador JDBC-ODBC.

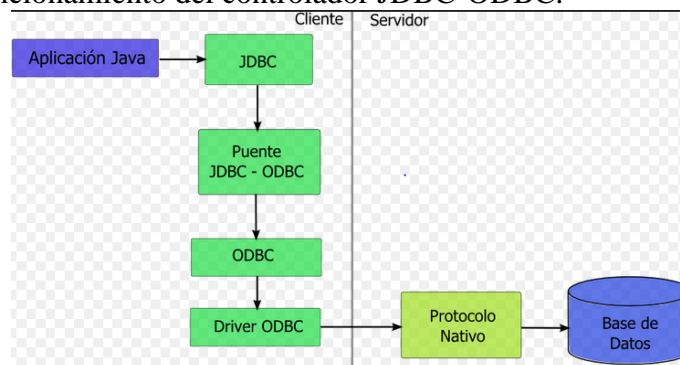


Figura 11. Esquema de funcionamiento del controlador JDBC-ODBC.

Imagen tomada del sitio Algo básico sobre Java

3.4.Lenguaje de programación

Antes de realizar el software se analizó qué lenguaje de programación será utilizado; de acuerdo a la aplicación del prototipo el software deberá tener la facilidad de poder ser reconocido en cualquier sistema operativo, sin necesidad de realizar cambios en el software.

3.4.1. Introducción al lenguaje de programación Java

El lenguaje de programación utilizado para este proyecto que puede ser llevado a cualquier sistema operativo es Java, se realizará una breve descripción del lenguaje de programación.

Para García de Jalón (2000) Java es un lenguaje de programación orientada a objetos, desarrollado por James Gosling y sus compañeros de Sun Microsystems al principio de la década de los 90. Sun Microsystems, pensó en crear un lenguaje que se basara en lenguajes de implementación que más se utilizaban en el mundo como son C y C++.

El lenguaje de programación Java no debe ser confundido con JavaScript ya que éste es un lenguaje interpretado, es decir, que no requiere compilación. La sintaxis de Java es similar a la utilizada en los lenguajes C y C++, diferenciándose por la sencillez de Java porque elimina herramientas de bajo nivel como es el caso de los punteros (pág. 160).

Java es un lenguaje que ha sido diseñado para producir software; presenta las siguientes características:

- **Confiable:** Minimiza los errores que se escapan a la fase de prueba.
- **Multiplataforma:** Una vez ya compilado el código Java puede llevarse a cualquier sistema operativo sin ser modificado y ejecutarlo allí sin problemas. Esto es porque el código se compila en un lenguaje intermedio llamado bytecodes que podrá ser leído independientemente de la máquina. Este lenguaje intermedio es interpretado por la Máquina Virtual Java (JVM) que será necesaria en la plataforma en la que se quiera ejecutar el código.
- **Seguro:** Applets recuperados por medio de la red no pueden causar daño a los usuarios.
- **Orientado a objetos:** Beneficioso tanto para el proveedor de bibliotecas de clases como para el programador de aplicaciones. Cuando se escriben programas en lenguajes orientados a objetos, no se definen objetos verdaderos sino se definen clases de objetos.
- **Robusto:** Los errores se detectan en el momento de producirse, lo que facilita la depuración.

Según García de Jalón (2000) Java es un lenguaje provisto de interfaces gráficos con el usuario, que permitirá aprovechar capacidades de multimedia de gráficos, imágenes, animación, audio e incluso video. Otra ventaja de Java es que posee muchas clases de los paquetes de la Java API (del inglés Application Programming Interface -Interfaz de Programación de Aplicaciones) que pueden ser reutilizados.

Con todo esto Java tiene un potencial para convertirse en un lenguaje de programación de aplicación general más importante del mundo (pág. 163).

Los sistemas Java se forman de las siguientes partes:

- **Un entorno:** El entorno utilizado para el proyecto es el Netbeans que es un entorno gratuito de código abierto para la generación de código en diversos lenguajes (especialmente pensado para Java). La descarga de este entorno se lo puede hacer desde la página web www.netbeans.org.
- **El Lenguaje:** El compilador Java y el intérprete Java son diferentes. El compilador se utiliza para los archivos fuente, a fin de crear archivos .class y el intérprete se usa para ejecutar los archivos de clase. Todo el código fuente Java se escribe en documentos de texto con extensión .java. Al ser un lenguaje para Internet, la codificación de texto debe permitir a todos los programadores de cualquier idioma escribir ese código. Eso significa que Java es compatible con la codificación Unicode.

En la práctica significa que los programadores que usen lenguajes distintos del inglés no tendrán problemas para escribir símbolos en su idioma, esto se puede extender para nombres de clase, variables, etc.

El compilador de Java no hace caso a los espacios, tabuladores y demás; en Java éstos solo sirven para mejorar la comprensión del programa. El compilador detectará los errores de sintaxis, en cambio los errores de lógica se perciben el momento de la ejecución; al momento de existir un error de lógica fatal produce que el programa termine antes de lo esperado. El error de lógica no fatal hará que el programa continúe su ejecución pero sin dar los resultados esperados.

El compilador de Java solo podrá evaluar expresiones en donde los tipos de datos sean idénticos.

3.4.2. Interfaz de persistencia Java (JPA) - entidades y managers

Java Persistence API (JPA) proporciona un modelo de persistencia basado en POJO's para mapear bases de datos relacionales en Java. La persistencia de Java fue desarrollada por expertos de EJB 3.0 como parte de JSR 220, aunque su uso no se limita a los componentes software EJB. Se puede utilizar en aplicaciones web y aplicaciones clientes.

Para ello, combina ideas y conceptos de los principales frameworks de persistencia, como Hibernate, Toplink y JDO. El mapeo objeto-relacional (es decir, la relación entre entidades Java y tablas de la base de datos, queries con nombre, etc.) se realiza mediante anotaciones en las propias clases de entidad. Pero para entender JPA, se tendrá que tener claro el concepto “persistencia”. La persistencia o el almacenamiento permanente, es una de las necesidades básicas de cualquier sistema de información de cualquier tipo. En primer lugar, se propuso que el programa tratara los datos haciendo consultas directas a la base de datos. Después, se propuso trabajar con objetos, pero las bases de datos tradicionales no admiten esta opción.

Debido a esta situación, aparecieron los motores de persistencia, cuya función es traducir entre los dos formatos de datos: de registros a objetos y de objetos a registros. Persistir objetos Java en una base de datos relacional implica serializar un árbol de objetos Java en una base de datos de estructura tabular y viceversa. Esencial es la necesidad de mapear objetos Java para optimizar velocidad y eficiencia de la base de datos.

i. Unidades de persistencia

La unidad de persistencia define un conjunto de todas las entidades (clases) que son gestionadas por la instancia del EntityManager en una aplicación. Este conjunto de clases de entidad representa los datos contenidos en una única BBDD.

Las unidades de persistencia se definen en el fichero de configuración persistence.xml. Aquí se muestra un ejemplo:

```
<persistence>
  <persistence-unit name="OrderManagement">
    <description>...</description>
    <jta-data-source>jdbc/MyOrderDB</jta-data-source>
    <jar-file>MyOrderApp.jar</jar-file>
    <class>com.widgets.Order</class>
    <class>com.widgets.Customer</class>
```

</persistence-unit>

</persistence>

Este archivo define una unidad de persistencia llamada OrderManagement Jar-file especifica los ficheros JAR en los que se encuentran las clases persistentes.

La persistencia puede tener 4 estados diferentes:

- Transient: Un objeto recién creado que no ha sido enlazado con el gestor de persistencia.
- Persistent: Un objeto enlazado con la sesión (Todos los cambios serán persistentes).
- Detached: Un objeto persistente que sigue en memoria después de que termina la sesión: existe en java y en la BDD.
- Removed: Un objeto marcado para ser eliminado de la BBDD: existe en java y se borrará de la BDD al terminar la sesión.

Modelo de relación persistencia-Java

<i>Modelo de objetos</i>	→	<i>Modelo Relacional</i>
Identidad	→	Clave primaria
Clase	→	Relación (Tabla, Vista)
Instancia	→	Fila
Atributo de instancia persistente con dominio un tipo básico o primitivo	→	Columna
Atributo persistente Referencia a objetos con tipo conocido	→	Clave Ajena
Atributo persistente tipo colección	→	No hay equivalencia directa, debe ser una relación, si la cardinalidad es (m:n), entonces una entidad de asociación; Si (1:n), entonces atributos en las relaciones correspondientes.
Atributo persistente de Clase	→	Una tabla con las características comunes a todas las instancias de una clase.
Herencia	→	Una relación agregada con todos los atributos de instancia de la jerarquía herencia completa o múltiples tablas

Figura 12. Modelo de relación persistencia-Java

Fuente: Algo básico sobre Java

ii. JPA - Entity Manager

Antes que nada hay que tener bien en claro dos temas muy importantes que son:

- Application-managed entity manager
- Container-managed entity manager

Siempre que una transacción sea iniciada, un nuevo contexto de persistencia (persistence context) es creado. Esto es así tanto para el Application-managed entity manager como también para el Container-managed entity manager: Para el caso del Application-managed entity manager (cuando no se usa un application server), la aplicación es la encargada de abrir y cerrar la transacción. Para el caso del Container managed entity manager (cuando se utiliza un ejb container), por defecto, la transacción es iniciada cuando se invoque desde el cliente al EJB. La transacción termina cuando finaliza la ejecución del método del session bean.

iii. Interfaces JPA

Los tipos de interfaces de las que se compone JPA son:

- **javax.persistence.Persistence:** Contiene métodos estáticos de ayuda para obtener una instancia de Entity Manager Factory de una forma independiente al vendedor de la implementación de JPA. Una clase de inicialización que va proporcionar un método estático para la creación de una Entity Manager Factory.
- **javax.persistence.EntityManagerFactory:** La clase `javax.persistence.Entity.Manager.Factory` nos ayuda a crear objetos de `EntityManager` utilizando el patrón de diseño del Factory. Este objeto en tiempo de ejecución representa una unidad de persistencia particular. Generalmente va a ser manejado como un singleton y proporciona métodos para la creación de instancias `EntityManager`.
- **javax.persistence.Entity:** La clase `javax.persistence.Entity` es una anotación Java que se coloca a nivel de clases Java serializables y que cada objeto de una de estas clases anotadas representa un registro de una base de datos.

- **javax.persistence.EntityManager:** Es la interfaz principal de JPA utilizada para la persistencia de las aplicaciones. Cada Entity Manager puede realizar operaciones (Create, Read, Update, Delete) sobre un conjunto de objetos persistentes. Es un objeto único, no compartido que representa una unidad de trabajo particular para el acceso a datos. Proporciona métodos para gestionar el ciclo de vida de las instancias entidad y para crear instancias Query.
- **javax.persistence.Query:** La interface javax.persistence.Query está implementada por cada vendedor de JPA para encontrar objetos persistentes manejando cierto criterio de búsqueda. JPA estandariza el soporte para consultas utilizando Java Persistence Query Language (JPQL) y Structured Query Language (SQL). Se puede obtener una instancia de Query desde una instancia de un Entity Manager.
- **javax.persistence.EntityTransaction:** Cada instancia de Entity Manager tiene una relación de uno a uno con una instancia de javax.persistence.EntityTransaction, permite operaciones sobre datos persistentes de manera que agrupados formen una unidad de trabajo transaccional, en el que todo el grupo sincroniza su estado de persistencia en la base de datos o todos fallan en el intento, en caso de fallo, la base de datos quedará con su estado original. Maneja el concepto de todos o ninguno para mantener la integridad de los datos.

iv. Entidades

Una entidad es un objeto de dominio de persistencia. Normalmente, una entidad representa una tabla en el modelo de datos relacional y cada instancia de esta entidad corresponde a un registro en esa tabla.

Las entidades podrán utilizar campos persistentes o propiedades. Si las anotaciones de mapeo se aplican a las instancias de las entidades, la entidad utiliza campos persistentes, En cambio, si se aplican a los métodos getters de la entidad, se utilizarán propiedades persistentes. Hay que tener en cuenta que no es posible aplicar anotaciones tanto a campos como a propiedades en una misma entidad.

Una entidad pasara a ser manejada por el contexto de persistencia de JPA cuando ésta sea persistida (mediante el método `persist()` del Entity Manager). En este punto, la entidad pasara a estar asociada a lo que comúnmente se le llama el contexto de persistencia. En este caso, y mientras la entidad sea manejada/asociada por el contexto de persistencia (también se las conoce como entidades atachadas o *attached entities*), el estado (valores de la propiedades) de la entidad será automáticamente sincronizado con la BD.

- **Campos de persistencia permanente:** si la entidad utiliza campos persistencia permanente, los accesos se realizan en tiempo de ejecución. Aquellos campos que no tienen anotaciones del tipo `javax.persistence.Transient` o no han sido marcados como Java transitorio serán persistentes para el almacenamiento de datos. Las anotaciones de mapeo objeto/relación deben aplicarse a los atributos de la instancia.
- **Propiedades de persistencia permanente:** si la entidad utiliza propiedades de persistencia permanente, la entidad debe seguir el método de los convenios de componentes JavaBeans. Las propiedades de JavaBean usan métodos getters y setters en cuyo nombre va incluido el atributo de la clase al cual hacen referencia. Si el atributo es booleano podrá utilizarse `isProperty` en lugar de `getProperty`.

v. Clases con claves primarias

Una clase con clave primaria debe cumplir los siguientes requerimientos:

- El modificador de control de acceso de la clase debe ser público
- Las propiedades de la clave primaria deben ser públicas o `protected` si se utiliza el acceso a la base de la propiedad.
- La clase debe tener un constructor público por defecto.
- La clase debe implementar los métodos `hashCode()` y `equals(Object other)`
- La clase debe ser serializable.

Una clave primaria debe representarse y mapearse por campos múltiples o

propiedades de la clase de la entidad, o debe representarse y mapearse como una clase embebida. Si la clave primaria está compuesta por varios campos o propiedades, los nombres y tipos de campos de la clave primaria o propiedades en la clave primaria debe coincidir con las de la entidad.

vi. Relaciones múltiples de la entidad

Hay cuatro tipos de relaciones: uno a uno, uno a muchos, muchos a uno, y muchos a muchos.

- Uno a uno: Cada entidad se relaciona con una sola instancia de otra entidad. Las relaciones uno a uno utilizan anotaciones de la persistencia de java “OneToOne”.
- Uno a muchos: Una entidad, puede estar relacionada con varias instancias de otras entidades. Las relaciones uno a muchos utilizan anotaciones de la persistencia de java “OneToMany” en los campos o propiedades persistentes.
- Muchos a uno: Múltiples instancias de una entidad pueden estar relacionadas con una sola instancia de otra entidad. Esta multiplicidad es lo contrario a la relación uno a muchos. Las relaciones muchos a uno utilizan anotaciones de la persistencia de java “ManyToOne” en los campos o propiedades persistentes.
- Muchos a muchos: En este caso varias instancias de una entidad pueden relacionarse con múltiples instancias de otras entidades. Este tipo de relación utiliza anotaciones de la persistencia de java “ManyToMany” en los campos o propiedades persistentes.

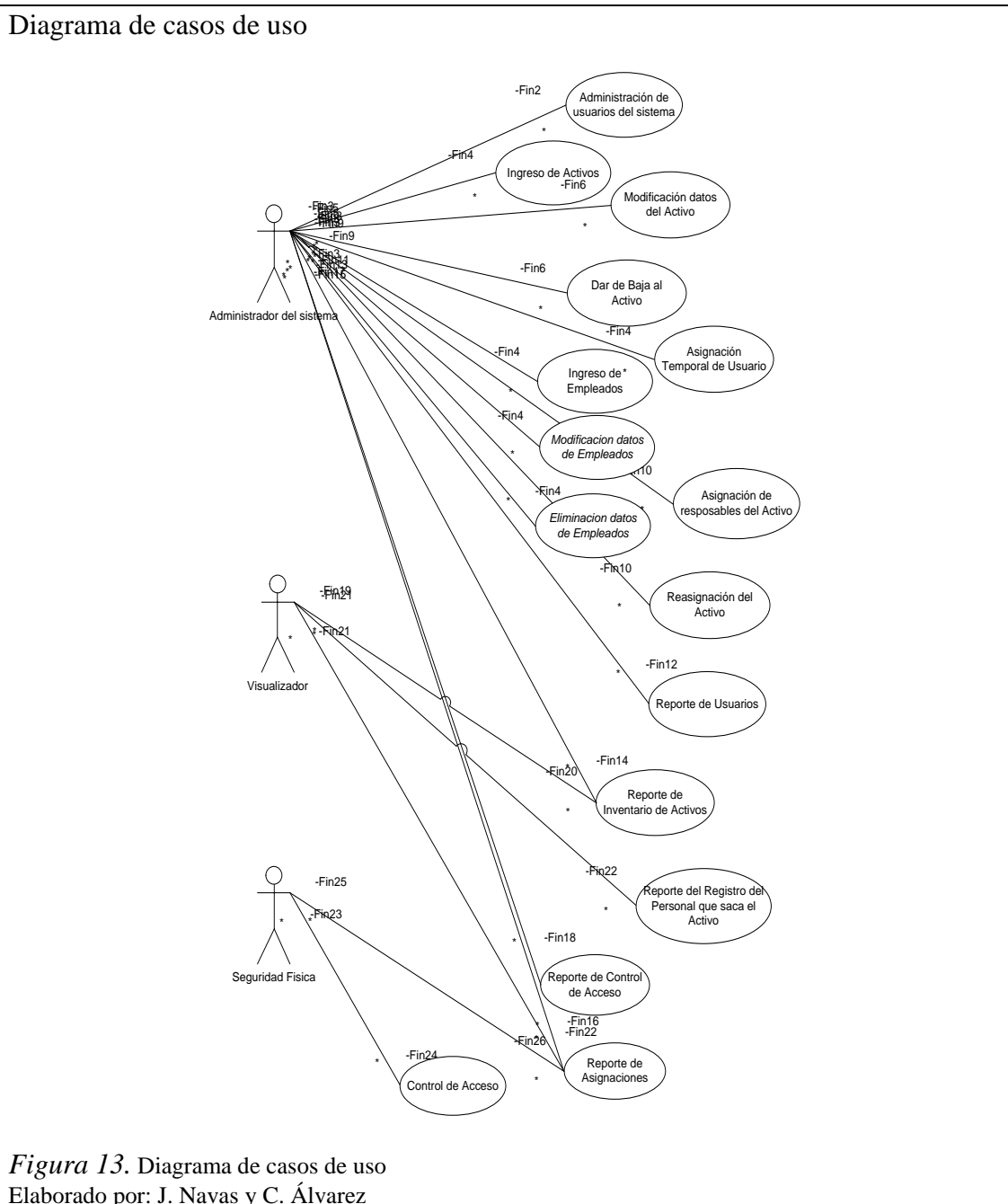
CAPÍTULO 4

DISEÑO E IMPLEMENTACIÓN

4.1. Diseño de la arquitectura del sistema

Para el diseño del sistema se ha tomado en cuenta las necesidades que han sido justificadas en el Capítulo II y por los diferentes departamentos involucrados mediante las guías de entrevista.

4.1.1. Diagrama de casos de uso



4.1.2.Documentación de casos de uso

Tabla 2. Documentación de casos de uso

Caso de uso	Documentación
Administración de usuarios del sistema	<p>Descripción: Permite crear a un usuario.</p> <p>Actores: administrador.</p> <p>Precondiciones: usuario no existe en el sistema.</p> <p>Flujo normal:</p> <ol style="list-style-type: none">1. Actor ingresa clave de usuario y datos.2. Presiona botón ingresar.3. Datos correctos, usuario se crea. <p>Flujo alternativo:</p> <ol style="list-style-type: none">4. Datos incorrectos. Se muestra mensaje de error, permitiendo la corrección de los datos. <p>Post condiciones: usuario existe en el sistema.</p>
Ingreso de activos	<p>Descripción: permite ingresar un nuevo activo.</p> <p>Actores: administrador.</p> <p>Precondiciones: activo no existe en el sistema.</p> <p>Flujo normal:</p> <ol style="list-style-type: none">1. Actor ingresa número de activo, tag y datos del equipo.2. Presiona botón ingresar.3. Datos correctos, producto se ingresa. <p>Flujo alternativo:</p> <ol style="list-style-type: none">4. Datos incorrectos. Se muestra mensaje de error, permitiendo la corrección de los datos.
Modificación de datos del activo	<p>Descripción: permite cambiar los datos de un activo.</p> <p>Actores: administrador.</p> <p>Precondiciones: activo existe en el sistema.</p> <p>Flujo normal:</p> <ol style="list-style-type: none">1. Ingresa número de activo del equipo para visualizar datos.2. Edita datos.3. Sistema comprueba validez de los datos y almacena en el sistema. <p>Flujo alternativo:</p> <ol style="list-style-type: none">4. Datos no son correctos. Se informa a usuario permitiendo corregirlos. <p>Post condiciones: nuevos datos almacenados en el sistema.</p>
Dar de baja al activo	<p>Descripción: permite dar de baja un activo. Actores: administrador.</p> <p>Precondiciones: activo existe en el sistema.</p> <p>Flujo normal:</p> <ol style="list-style-type: none">1. Ingresa número de activo del equipo.2. Sistema valida datos y lo deshabilita. <p>Flujo alternativo:</p> <ol style="list-style-type: none">3. Número de activo no es correcto o no existe. Se informa a Administrador, permitiendo corregir los datos. <p>Post condiciones: el registro de equipo deshabilitado en el sistema.</p>

Ingreso de empleados	<p>Descripción: permite crear a un empleado.</p> <p>Actores: administrador.</p> <p>Precondiciones: empleado no existe en el sistema.</p> <p>Flujo normal:</p> <ol style="list-style-type: none"> 1. Actor ingresa cedula de empleado y datos. 2. Presiona botón ingresar. 3. Datos correctos, empleado se da ingreso. <p>Flujo alternativo:</p> <ol style="list-style-type: none"> 4. Datos incorrectos. Se muestra mensaje de error, permitiendo la corrección de los datos.
Modificación datos de empleado	<p>Descripción: permite cambiar los datos de un empleado.</p> <p>Actores: administrador.</p> <p>Precondiciones: empleado ya existe en el sistema.</p> <p>Flujo normal:</p> <ol style="list-style-type: none"> 1. Ingresa n° de cédula del empleado para visualizar datos. 2. Edita datos. 3. Sistema comprueba validez de los datos y almacena en el sistema. <p>Flujo alternativo:</p> <ol style="list-style-type: none"> 4. Datos no son correctos. Se muestra un mensaje permitiendo corregirlos. <p>Post condiciones: nuevos datos almacenados en el sistema.</p>
Dar de baja empleado	<p>Descripción: permite dar de baja un empleado.</p> <p>Actores: administrador.</p> <p>Precondiciones: empleado ya existe en el sistema.</p> <p>Flujo normal:</p> <ol style="list-style-type: none"> 1. Ingresa n° de cédula del empleado. 2. Sistema valida cédula del empleado, previa notificación de talento humano mediante orden de salida y da de baja. <p>Flujo alternativo:</p> <ol style="list-style-type: none"> 3. Cedula de empleado no es correcta, o empleado no existe. Se muestra un mensaje, permitiendo corregir los datos. <p>Post condiciones: empleado deshabilitado en el sistema.</p>
Asignación temporal de usuario	<p>Descripción: permite crear a un usuario temporal.</p> <p>Actores: administrador.</p> <p>Precondiciones: empleado no se encuentra activo o no existe en el sistema.</p> <p>Flujo normal:</p> <ol style="list-style-type: none"> 1. Actor ingresa datos del usuario y características del equipo. 2. Asigna un tag temporal tipo llavero. 3. Se realiza una validación de los datos mediante la acción del botón ingresar. 4. Datos correctos, empleado queda registrado en usuarios temporales. <p>Flujo alternativo:</p> <ol style="list-style-type: none"> 5. Datos incorrectos. Se muestra mensaje de error, permitiendo la corrección de los datos

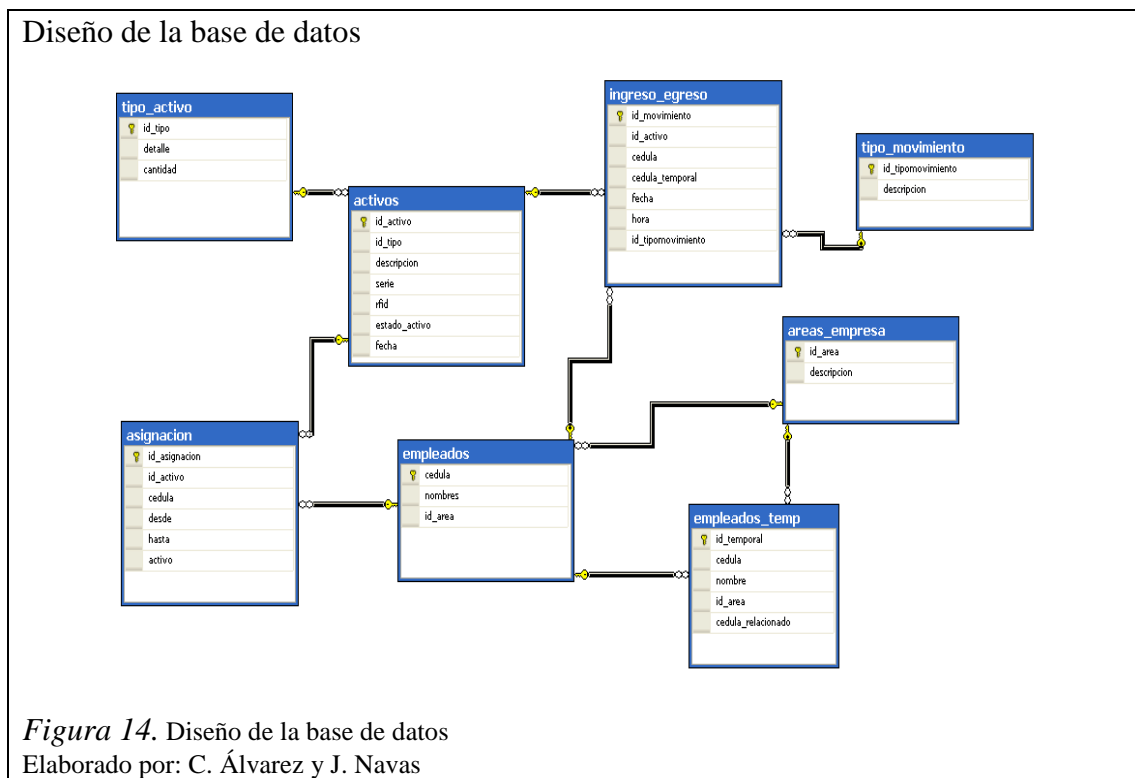
Asignación de responsables del activo	<p>Descripción: permite asignar un activo nuevo a un empleado.</p> <p>Actores: administrador.</p> <p>Precondiciones: empleado a quien se va a realizar la asignación existe en el sistema. Activo que se va a asignar existe en el sistema.</p> <p>Flujo normal:</p> <ol style="list-style-type: none"> 1. Actor ingresa empleado, número de activo de equipo y datos. 2. Presiona botón asignar y se validan los datos. 3. Datos correctos, asignación registrada. <p>Flujo alternativo:</p> <ol style="list-style-type: none"> 4. Datos incorrectos. Se muestra mensaje de error, permitiendo la corrección de los datos <p>Post condiciones: salida del empleado con el activo previa autorización documentada.</p>
Reasignación del activo	<p>Descripción: permite cambiar los datos del activo asignado a un empleado.</p> <p>Actores: administrador.</p> <p>Precondiciones: asignación existe y se encuentre activa en el sistema.</p> <p>Flujo normal:</p> <ol style="list-style-type: none"> 1. Consulta n° de cédula de empleado para visualizar sus asignaciones. 2. Se habilita la edición del campo empleado responsable del activo. 3. Se editan los datos. 4. Sistema comprueba validez de los datos y almacena en el sistema. <p>Flujo alternativo:</p> <ol style="list-style-type: none"> 5. Datos no son correctos. Se informa al actor permitiendo corregirlos. <p>Post condiciones: nuevos datos almacenados en el sistema.</p>
Reporte usuarios	<p>Descripción: permite generar y ver un reporte de los usuarios existentes en el sistema.</p> <p>Actores: administrador.</p> <p>Precondiciones: usuarios existentes.</p> <p>Flujo normal:</p> <ol style="list-style-type: none"> 1. Actor ingresa parámetros para generar reporte. 2. Presiona botón generar reporte. 3. Se muestra reporte basado en parámetros ingresados por el actor. Se da la opción de imprimir reporte. <p>Flujo alternativo:</p> <ol style="list-style-type: none"> 4. Parámetros ingresados por actor no coinciden con ningún registro. Se muestra reporte en blanco.
Reporte de inventario de activos	<p>Descripción: permite generar y ver un reporte de los activos existentes en el sistema.</p> <p>Actores: administrador, visualizador.</p> <p>Precondiciones: activos existentes.</p> <p>Flujo normal:</p> <ol style="list-style-type: none"> 1. Actor ingresa parámetros para generar reporte. 2. Presiona botón generar reporte. 3. Se muestra reporte basado en parámetros ingresados por el actor. Se da la opción de imprimir reporte. <p>Flujo alternativo:</p> <ol style="list-style-type: none"> 4. Parámetros ingresados por actor no coinciden con ningún registro. Se muestra reporte en blanco.

Reporte de control de acceso	<p>Descripción: permite generar y ver un reporte del control de acceso al sistema.</p> <p>Actores: administrador.</p> <p>Precondiciones: registros de control de acceso existentes. Flujo normal:</p> <ol style="list-style-type: none"> 1. Actor ingresa parámetros para generar reporte. 2. Presiona botón generar reporte. 3. Se muestra reporte basado en parámetros ingresados por el actor. Se da la opción de imprimir reporte. <p>Flujo alternativo:</p> <ol style="list-style-type: none"> 4. Parámetros ingresados por actor no coinciden con ningún registro. Se muestra reporte en blanco.
Reporte asignaciones	<p>Descripción: permite generar y ver un reporte de las asignaciones existentes en el sistema.</p> <p>Actores: administrador, visualizador, seguridad.</p> <p>Precondiciones: asignaciones existentes.</p> <p>Flujo normal:</p> <ol style="list-style-type: none"> 1. Actor ingresa parámetros para generar reporte. 2. Presiona botón generar reporte. 3. Se muestra reporte basado en parámetros ingresados por el actor. Se da la opción de imprimir reporte. <p>Flujo alternativo:</p> <ol style="list-style-type: none"> 4. Parámetros ingresados por actor no coinciden con ningún registro. Se muestra reporte en blanco.
Reporte del registro del personal que saca el equipo	<p>Descripción: permite generar y ver un reporte del personal que ha salido junto con un activo informático, que no está a su cargo.</p> <p>Actores: administrador, visualizador, seguridad.</p> <p>Precondiciones: asignaciones existentes.</p> <p>Flujo normal:</p> <ol style="list-style-type: none"> 1. Actor ingresa parámetros para generar reporte. 2. Presiona botón generar reporte. 3. Se muestra reporte basado en parámetros ingresados por el actor. Se da la opción de imprimir reporte. <p>Flujo alternativo:</p> <ol style="list-style-type: none"> 4. Parámetros ingresados por actor no coinciden con ningún registro. Se muestra reporte en blanco.
Control de acceso	<p>Descripción: realiza el control de acceso.</p> <p>Actores: seguridad.</p> <p>Precondiciones: usuarios, equipos y asignaciones ingresados en Sistema.</p> <p>Flujo normal:</p> <ol style="list-style-type: none"> 1. Lector rfid realiza lectura automáticamente cada cierto tiempo en los diferentes accesos de la empresa. 1. Sistema lee resultados de la lectura rfid. 2. Muestra que fue lo que cruzó por el lector, y si tiene permisos para entrar o salir. 3. Registra acción en control acceso. <p>Flujo alternativo:</p> <ol style="list-style-type: none"> 1. Usuario puede decidir en qué momento realizar otra lectura. <p>Post condiciones: registro en control acceso.</p>

Nota: Documentación de casos de uso, adaptado de la investigación, por J. Navas & C. Álvarez

4.2. Diseño de la base de datos

El Diseño de la base de datos ha sido creado luego de un análisis de los requerimientos del sistema como son: la descripción de los activos, datos de los empleados, tipos de activos, áreas existentes en la empresa, asignación de los activos a los empleados, movimientos de los activos, empleados temporales que ingresan activos en la empresa, perfiles de acceso al sistema, y reportes de inventario (ver Figura 14).



4.2.1. Descripción de tablas

- Tabla tipo_activo
 - Clave primaria: id_tipo
 - Campo 1: detalle
 - Campo 2: cantidad
 - Relación: 1 a varios tabla activos
- Tabla asignación
 - Clave primaria: id_asignacion
 - Campo 1: id_activo
 - Campo 2: cedula

- Campo 3: desde
 - Campo 4: hasta
 - Campo 5: activo
 - Relación: varios a 1 tabla activos
- Tabla activos
 - Clave primaria: id_activo
 - Campo 1: id_tipo
 - Campo 2: descripcion
 - Campo 3: serie
 - Campo 4: rfid
 - Campo 5: estado_activo
 - Campo 6: fecha
 - Relación 1: varios a 1 tabla tipo_activo
 - Relación 2: 1 a varios tabla asignacion
 - Relación 3: 1 a varios tabla ingreso_egreso
- Tabla ingreso_egreso
 - Clave primaria: id_movimiento
 - Campo 1: id_activo
 - Campo 2: cedula
 - Campo 3: cedula_temporal
 - Campo 4: fecha
 - Campo 5: hora
 - Campo 6: id_tipomovimiento
 - Relación 1: varios a 1 tabla activos
 - Relación 2: varios a 1 tabla empleados
 - Relación 3: varios a 1 tabla tipo_movimiento
- Tabla tipo_movimiento
 - Clave primaria: id_tipomovimiento
 - Campo 1: descripcion
 - Relación 1: 1 a varios tabla ingreso_egreso
- Tabla areas_empresa
 - Clave primaria: id_area
 - Campo 1: descripcion
 - Relación 1: 1 a varios tabla empleados
 - Relación 2: 1 a varios tabla empleados_temp

- Tabla empleados_temp
 - Clave primaria: id_temporal
 - Campo 1: cedula
 - Campo 2: nombre
 - Campo 3: id_area
 - Campo 4: cedula_relacionado
 - Relación 1: varios a 1 tabla áreas_empresa
 - Relación 2: varios a 1 tabla empleados
- Tabla empleados
 - Clave primaria: cedula
 - Campo 1: nombres
 - Campo 2: id_area
 - Relación 1: 1 a varios tabla asignacion
 - Relación 2: 1 a varios tabla empleados_temp
 - Relación 3: 1 a varios tabla ingreso_egreso
 - Relación 4: varios a 1 tabla áreas_empresa

4.3. Diseño del sistema

El sistema fue desarrollado en la plataforma Netbeans basado en el lenguaje de programación Java mediante la metodología XP, con un modelo cliente servidor, el cual almacena su información en una base de datos SQL Server 2008.

4.3.1. Creación de la persistencia en el sistema

Para crear una persistencia en el sistema se realizan los siguientes pasos:

1. Para iniciar la configuración de JPA en nuestro proyecto lo primero que se debe hacer es crear el archivo persistence.xml, para hacer esto se realiza lo siguiente: Clic derecho sobre el proyecto, New -> Other -> persistence->Persistence Unit (ver Figura 15).

Creación de persistencia

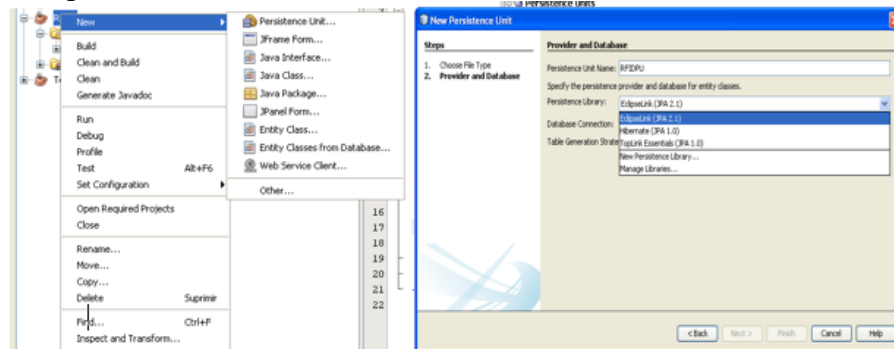


Figura 15. Creación de persistencia

Elaborado por: C. Álvarez y J. Navas

2. Se procede a importar la librería de persistencia en nuestro caso será Toplink Essentials la que utilizará (ver Figura 16).

Insertión de la librería Toplink

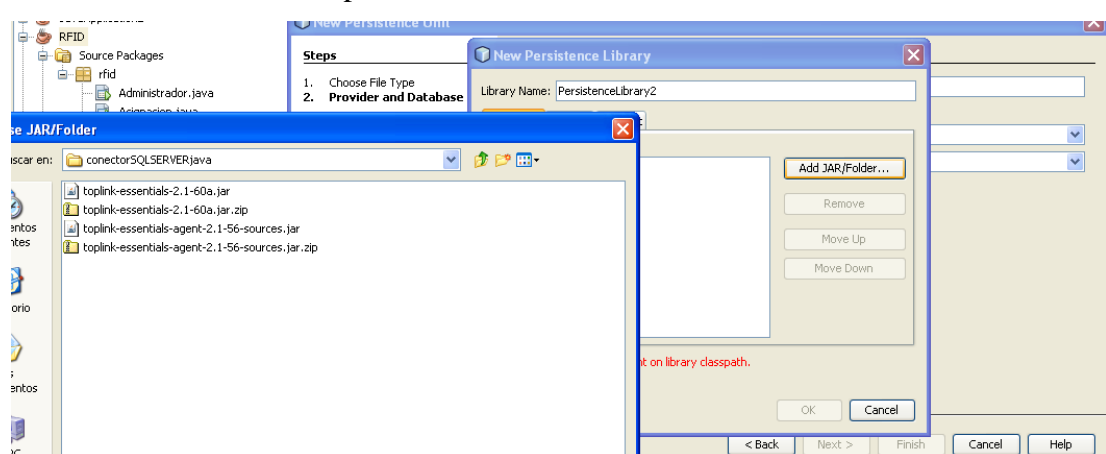


Figura 16. Inserción de la librería Toplink

Imagen elaborada por: C. Álvarez y J. Navas

3. Proceder a crear una nueva conexión con la librería Toplink ya elegida anteriormente (ver Figura 17).

Creación de conexión a la base de datos

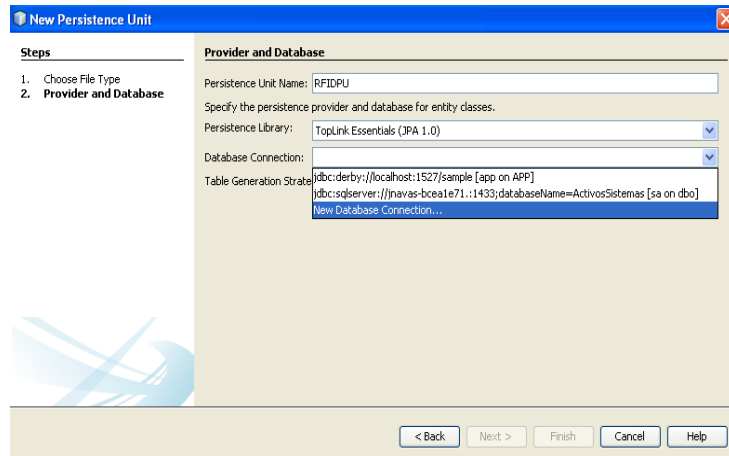


Figura 17. Creación de conexión a la base de datos

Imagen elaborada por: Carlos Álvarez

4. Se procede a cargar el driver que se utilizará para la conexión (ver Figura 18).

Carga de driver para SQL

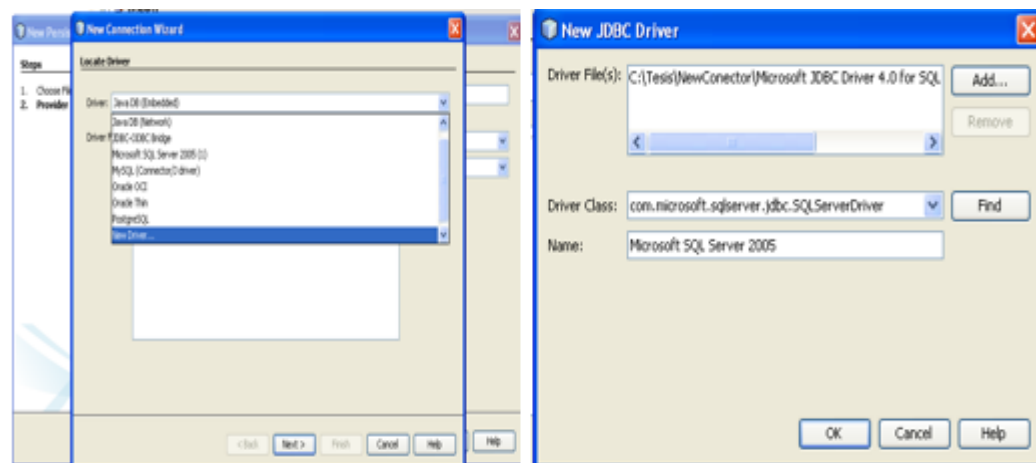


Figura 18. Carga de driver para SQL

Imagen elaborada por: Carlos Álvarez

5. Se configuran los parámetros de la base de datos como indican los campos (ver Figura 19).

Configuración de parámetros SQL

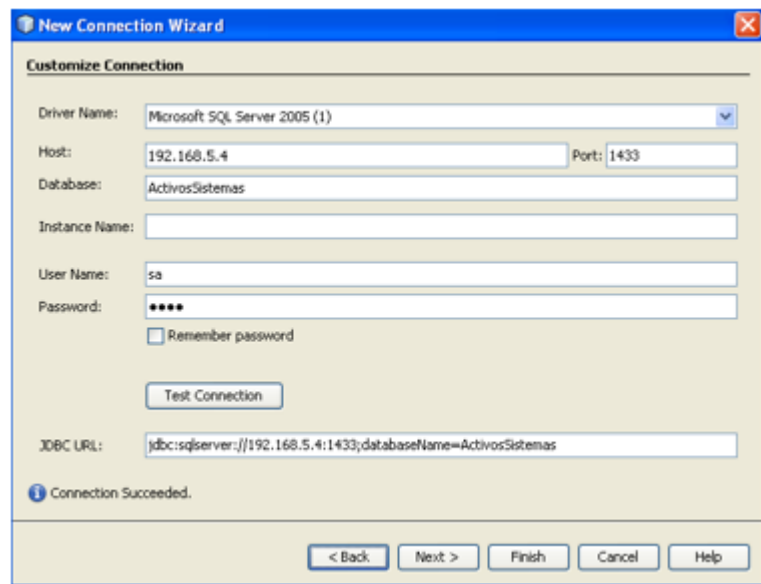


Figura 19. Configuración de parámetros SQL

Elaborado por: C. Álvarez y J. Navas

6. Escoger el nivel de acceso a la Base de Datos (ver Figura 20), para este caso dbo.

Nivel de acceso SQL

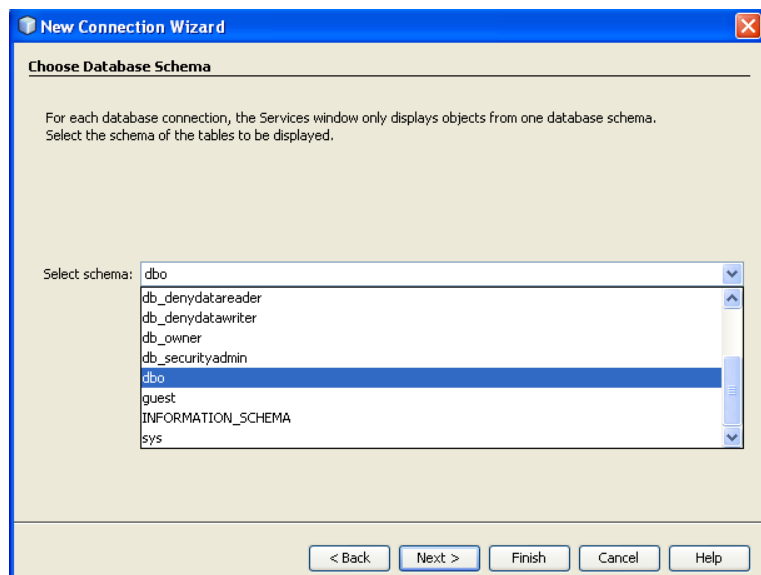


Figura 20. Nivel de acceso SQL

Elaborado por: C. Álvarez y J. Navas

7. Dar Clic en Finish para completar la creación de la conexión (ver Figura 21).

Verificación de la conexión creada

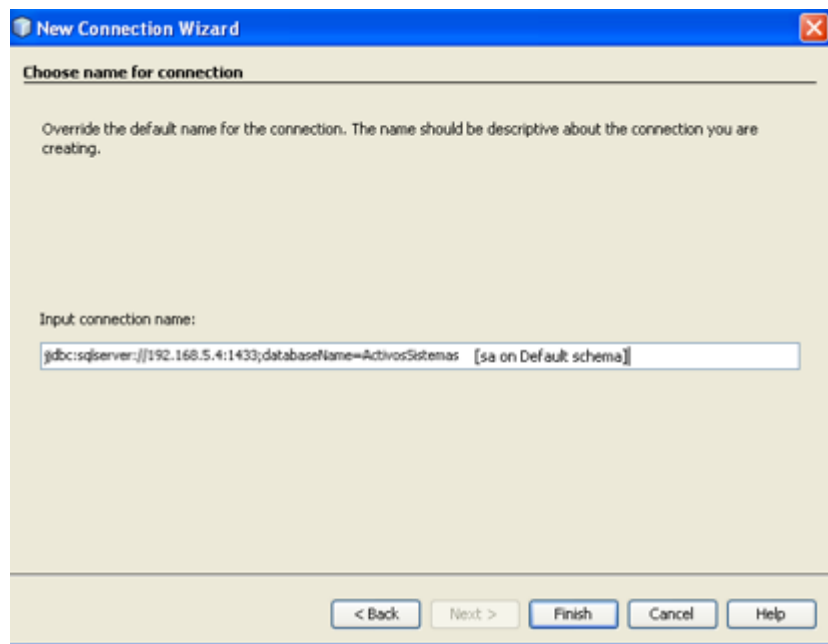


Figura 21. Verificación de la conexión creada

Elaborado por: C. Álvarez y J. Navas

8. Por último dar Clic en Finish para terminar la creación de la Persistencia (ver Figura 22).

Finalización de persistencia

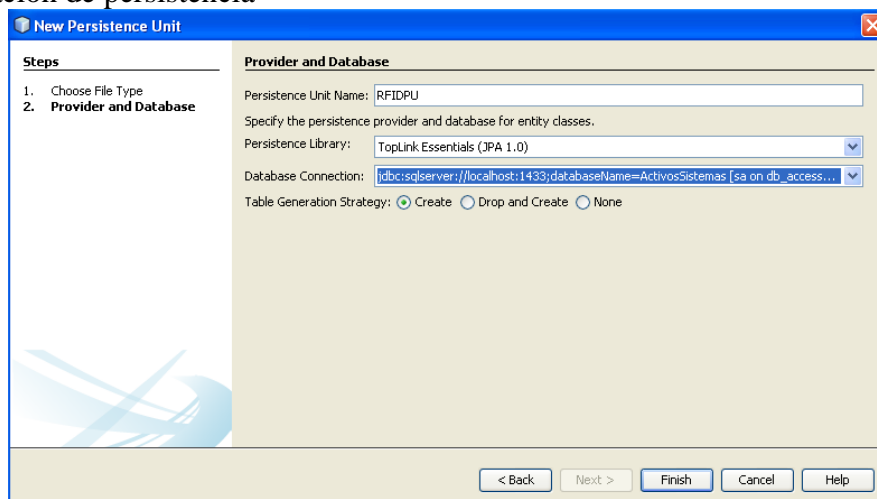


Figura 22. Finalización de persistencia

Imagen elaborada por: Carlos Álvarez

9. Luego de configurar la unidad de persistencia a través de Netbeans, nos agregara en las librerías todo lo necesario para usar JPA con Toplink, además crea la Carpeta META-INF y le anexa el archivo de configuración persistence.xml (ver Figura 23).

Persistencia creada



Figura 23. Persistencia creada

Elaborado por: C. Álvarez y J. Navas

10. Proceder a crear una Entidad de Persistencia dando Clic en: New -> Other -> persistence->EntityClasses from Database (ver Figura 24).

Creación de entidad de persistencia

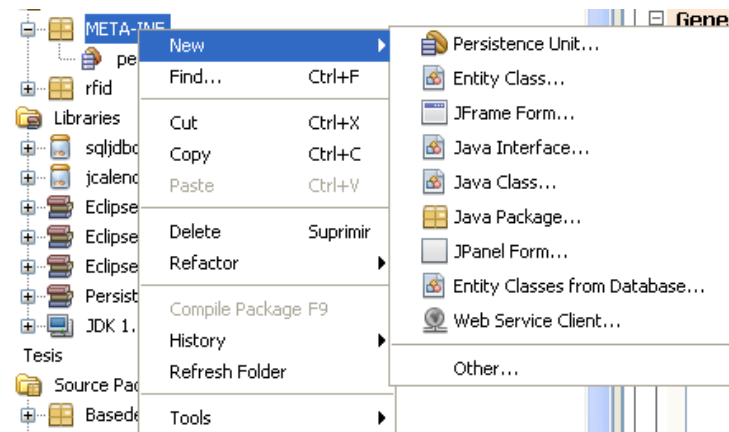


Figura 24. Creación de entidad de persistencia

Imagen elaborada por: Carlos Álvarez

11. Se presentaran las Tablas de nuestra base de datos, añadimos todas las tablas para crear las entidades (ver Figura 25).

Selección de tablas en la entidad de persistencia

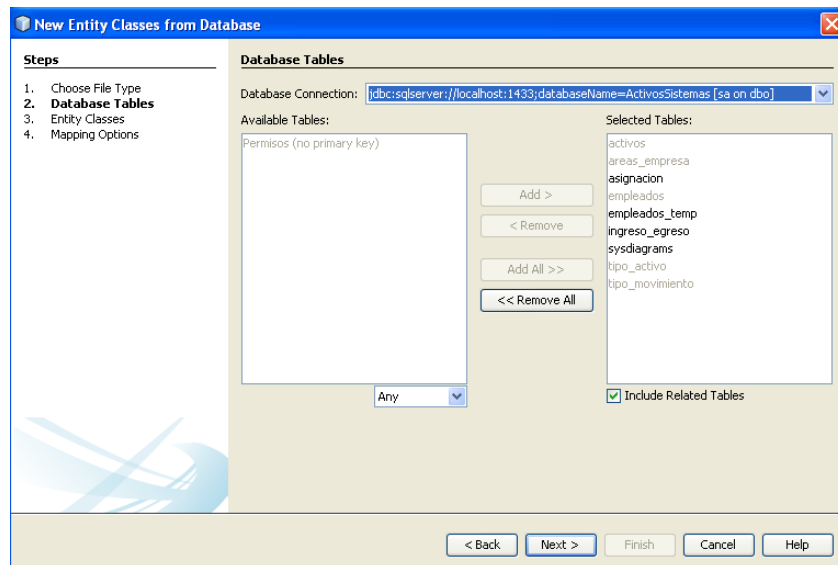


Figura 25. Selección de tablas en la entidad de persistencia

Imagen elaborada por: Carlos Álvarez

12. Escoger el paquete donde se crearan las entidades, siempre debe ser un paquete diferente en el que la persistencia esta creada (ver Figura 26).

Selección de paquete

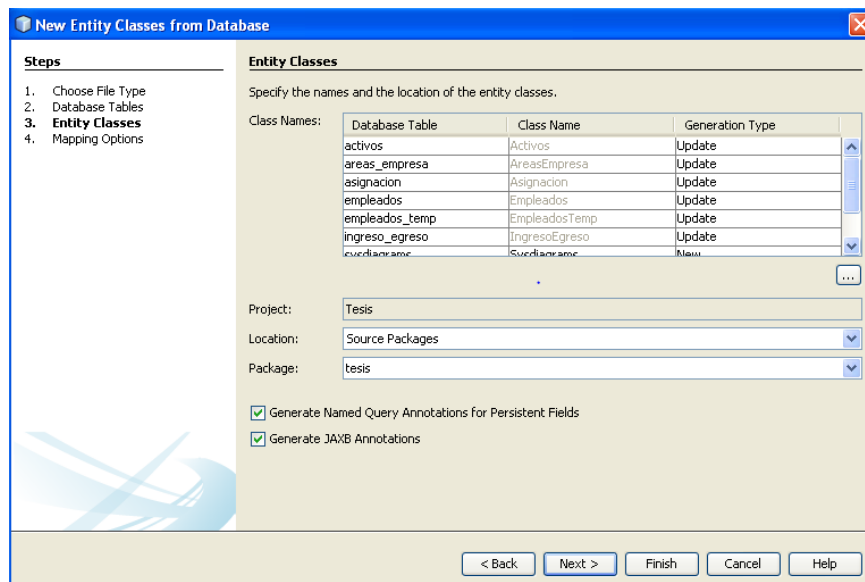


Figura 26. Selección de paquete

Elaborado por: C. Álvarez y J. Navas

13. Finaliza la creación de las Entidades dando Clic en Finish (ver Figura 27).

Finalización de entidades

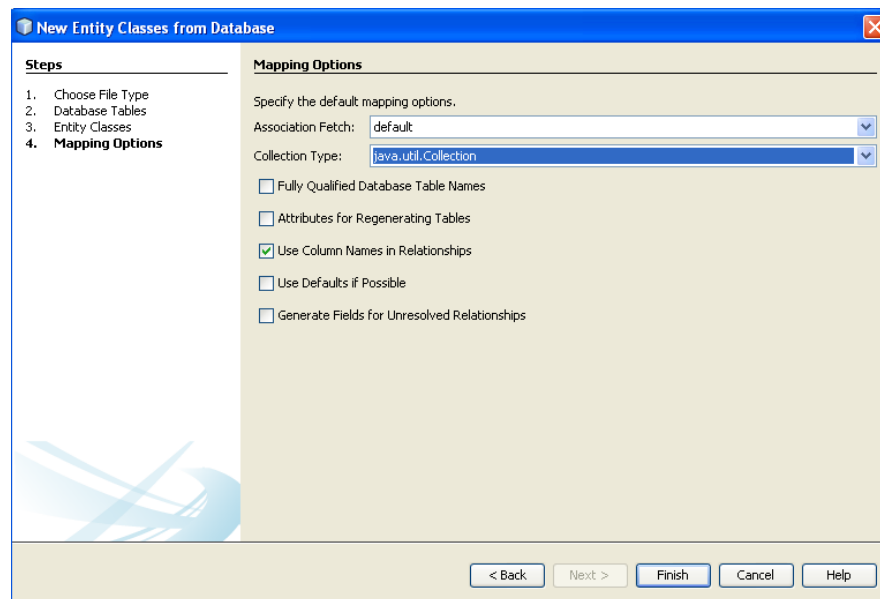


Figura 27. Finalización de entidades

Elaborado por: C. Álvarez y J. Navas

14. Se verifican que las entidades fueron creadas en el paquete de manera automática, importando las tablas de la base de datos hacia clases (ver Figura 28).

Comprobación de clases de entidades

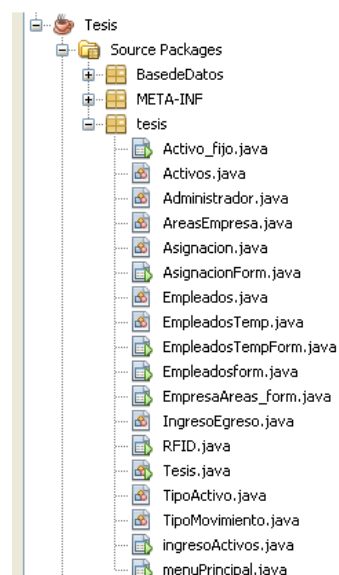


Figura 28. Comprobación de clases de entidades

Elaborado por: C. Álvarez y J. Navas

15. El siguiente código muestra la persistencia que permite la conectividad con la base de datos de manera directa mediante la tecnología Toplink, el cual se genera automáticamente un archivo con extensión XML.

```
<?xml version="1.0" encoding="UTF-8"?>
<persistence version="1.0" xmlns="http://java.sun.com/xml/ns/persistence"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://java.sun.com/xml/ns/persistence
http://java.sun.com/xml/ns/persistence/persistence_1_0.xsd">
<persistence-unit name="TesisPU" transaction-
type="RESOURCE_LOCAL">
<provider>oracle.toplink.essentials.PersistenceProvider</provider>
<class>tesis.IngresoEgreso</class>
<class>tesis.Activos</class>
<class>tesis.TipoActivo</class>
<class>tesis.TipoMovimiento</class>
<class>tesis.EmpleadosTemp</class>
<class>tesis.Asignacion</class>
<class>tesis.AreasEmpresa</class>
<class>tesis.Empleados</class>
<properties>
<property name="toplink.jdbc.user" value="sa"/>
<property name="toplink.jdbc.password" value="sa1"/>
<property name="toplink.jdbc.url"
value="jdbc:sqlserver://192.168.5.4:1433;databaseName=ActivosSistemas"/>
<property name="toplink.jdbc.driver"
value="com.microsoft.sqlserver.jdbc.SQLServerDriver"/>
</properties>
</persistence-unit>
</persistence>
```

4.3.2. Manejo de la información de la base de datos

1. Las clases creadas por la importación de las tablas de la base de datos son constituidas por métodos y variables, los cuales permiten la interacción entre el programa y la base de datos.

Las sentencias de consulta SQL son creadas de manera automática por cada campo de cada tabla y asignadas a una variable por el paquete de persistencia del programa (ver Figura 29).

Creación de sentencias SQL

```
@Entity
@Table(name = "activos")
@XmlRootElement
@NamedQueries({
    @NamedQuery(name = "Activos.findAll", query = "SELECT a FROM Activos a"),
    @NamedQuery(name = "Activos.findByIdactivo", query = "SELECT a FROM Activos a WHERE a.idactivo = :idactivo"),
    @NamedQuery(name = "Activos.findByDescripcion", query = "SELECT a FROM Activos a WHERE a.descripcion = :descripcion"),
    @NamedQuery(name = "Activos.findBySerie", query = "SELECT a FROM Activos a WHERE a.serie = :serie"),
    @NamedQuery(name = "Activos.findByRfid", query = "SELECT a FROM Activos a WHERE a.rfid = :rfid"),
    @NamedQuery(name = "Activos.findByEstadoactivo", query = "SELECT a FROM Activos a WHERE a.estadoactivo = :estadoactivo"),
    @NamedQuery(name = "Activos.findByFecha", query = "SELECT a FROM Activos a WHERE a.fecha = :fecha"))
public class Activos implements Serializable {
    private static final long serialVersionUID = 1L;
    @Id
    @Basic(optional = false)
    @Column(name = "id_activo")
    private Integer idactivo;
    @Column(name = "descripcion")
    private String descripcion;
    @Column(name = "serie")
    private String serie;
    @Column(name = "rfid")
    private String rfid;
    @Column(name = "estado_activo")
    private Boolean estadoactivo;
    @Column(name = "fecha")
    @Temporal(TemporalType.DATE)
    private Date fecha;
    @JoinColumn(name = "id_tipo", referencedColumnName = "id_tipo")
    @ManyToOne
```

Figura 29. Creación de sentencias SQL

Imagen elaborada por: C. Álvarez y J. Navas

2. Por cada campo de cada tabla también es creado un método set y get, permitiendo así la inserción o actualización de un campo utilizando el método set y la función de consulta si se hace uso del método get. (ver Figura 30).

Creación de métodos set y get

```
public Activos() {  
}  
  
public Activos(Integer idActivo) {  
    this.idActivo = idActivo;  
}  
  
public Integer getIdActivo() {  
    return idActivo;  
}  
  
public void setIdActivo(Integer idActivo) {  
    this.idActivo = idActivo;  
}  
  
public String getDescripcion() {  
    return descripcion;  
}  
  
public void setDescripcion(String descripcion) {  
    this.descripcion = descripcion;  
}
```

Figura 30. Creación de métodos set y get
Elaborado por: C. Álvarez y J. Navas

3. Para la utilización de los métodos set y get, se ha creado los métodos de Grabar, Modificar y Eliminar, las cuales se utilizan a lo largo del programa estas se encuentran dentro de la clase **Administrador**.

4.3.2.1. Proceso para insertar información a la base de datos

Seguimos los siguientes pasos:

- i) En cada botón con la opción de Grabar o Insertar se tienen instanciadas las clases relacionadas con las tablas de la Base de Datos en las cuales utiliza los métodos set y get y la clase **Administrador** donde se ubica el método **Grabar**.
- ii) Se hace el llamado al método set para realizar el envío de datos al campo de la tabla al cual se quiere realizar la inserción.
- iii) Luego se invoca al método **Grabar** y pasa al método una variable de tipo objeto la cual contiene las sentencias con la información de los campos de la tabla para realizar la inserción de los datos (ver Figura 31).

Inserción de datos a la base

```
private void jButton1ActionPerformed(java.awt.event.ActionEvent evt) {  
    // TODO add your handling code here:  
    int clave;  
    if(!textdescripcion.getText().isEmpty())  
    {  
        try  
        {  
            areas = new AreasEmpresa();  
            admin = new Administrador();  
            clave = admin.getNuevaClave("AreasEmpresa", "idArea");  
            areas.setDescription(textdescripcion.getText());  
            areas.setIdArea(clave);  
            admin.Grabar(areas);  
            JOptionPane.showMessageDialog(this, "Área ingresada exitosamente", "Ingreso de datos", 1);  
        }  
        catch(Exception e){  
            System.out.println("Error de Insercion " + e);  
        }  
    }  
    else  
    {  
        JOptionPane.showMessageDialog(this, "El campo descripción es obligatorio", "Ingreso de datos", 0);  
    }  
    this.cargarTabla();  
}
```

Figura 31. Inserción de datos a la base
Elaborado por: C. Álvarez y J. Navas

- iv) El método **Grabar** de la Clase Administrador, hace un llamado a la clase **EntityManager** del paquete de persistencia, la cual nos permite realizar la operación de inserción sobre la base de datos, el ciclo de vida de la instancia consiste en realizar un llamado al método `begin()` que inicializa la transacción, luego continua con el envío del objeto persistente mediante el método `persist()` que permite la creación de un nuevo objeto y termina con el método `commit()` el cual confirma la inserción de los datos (ver Figura 32).

Método grabar

```
public void Grabar(Object object) {  
    // if(log.isDebugEnabled()) log.debug("grabarObjeto() llamado");  
    //EntityManagerFactory emf = Persistence.createEntityManagerFactory("T  
    EntityManager em;  
    em = getEMF().createEntityManager();  
    try {  
        em.getTransaction().begin();  
        em.persist(object);  
        em.getTransaction().commit();  
        //System.out.println("REGISTRO ALMACENADO CON EXITO");  
    } catch (Exception e) {  
        System.out.println("ERROR AL GRABAR" + e.getStackTrace());  
    }  
}
```

Figura 32. Método grabar
Elaborado por: C. Álvarez y J. Navas

4.3.2.2. Proceso para modificar información a la base de datos

Seguimos los siguientes pasos:

- i) En cada botón con la opción de Modificar o Actualizar se tienen instanciadas las clases relacionadas con las tablas de la Base de Datos en las cuales utiliza los métodos set y get y la clase **Administrador** donde se ubica el método **Modificar**.
- ii) Se hace el llamado al método set para realizar el envío de datos al campo de la tabla al cual se desea realizar la modificación.
- iii) Luego invoca al método **Modificar** y pasa al método una variable de tipo objeto la cual contiene las sentencias con la información de los campos de la tabla para realizar la modificación de los datos (ver Figura 33).

Actualización de datos a la base

```
private void jButton2ActionPerformed(java.awt.event.ActionEvent evt) {  
    // TODO add your handling code here:  
    AreasEmpresa ar = new AreasEmpresa();  
    if(seleccioareas != null)  
    {  
        try  
        {  
            seleccioareas.setDescripcion(textdescripcion.getText());  
            admin.Modificar(seleccioareas);  
            JOptionPane.showMessageDialog(this, "Area actualizado exitosamente", "Actualización de datos", 1);  
            seleccioareas = null;  
        }  
        catch(Exception e){  
            System.out.println("Error de Actualización " + e);  
        }  
    }  
    else  
    {  
        JOptionPane.showMessageDialog(this, "Seleccione un área para actualizar", "Error al actualizar", 0);  
    }  
    this.cargarTabla();  
}
```

Figura 33. Actualización de datos a la base

Elaborado por: C. Álvarez y J. Navas

- iv) El método **Modificar** de la Clase Administrador, hace un llamado a la clase **EntityManager** del paquete de persistencia, la cual nos permite realizar la operación de modificación sobre la base de datos, el ciclo de vida de la instancia consiste en realizar un llamado al método begin() que inicializa la transacción, luego continua con el envío del objeto persistente mediante el método merge() que actualiza el objeto y termina con el

método commit() el cual confirma la modificación de los datos (ver Figura 34).

Método modificar

```
public void Modificar(Object object) {  
    //      if(log.isDebugEnabled()) log.debug("grabarObjeto() llamado");  
    //EntityManagerFactory emf = Persistence.createEntityManagerFactory("InvConexionPU");  
    //Integer clave = 0;  
    EntityManager em = getEMF().createEntityManager();  
    try {  
        em.getTransaction().begin();  
        object = em.merge(object);  
        em.getTransaction().commit();  
        //System.out.println("rREGISTRO ALMACENADO CON EXITO");  
    } catch (Exception e) {  
        System.out.println("ERROR AL MODIFICAR" + e);  
    }  
}
```

Figura 34. Método modificar

Elaborado por: C. Álvarez y J. Navas

4.3.2.3. Proceso para eliminar información a la base de datos

Seguimos los siguientes pasos:

- i) En cada botón con la opción de Borrar o Eliminar se tienen instanciadas las clases relacionadas con las tablas de la Base de Datos en las cuales utiliza los métodos set y get y la clase **Administrador** donde se ubica el método **Eliminar**.
- ii) Se hace el llamado al método set para realizar el envío de datos al campo de la tabla al cual se desea realizar la eliminación.
- iii) Luego invoca al método **Eliminar** y pasa al método una variable de tipo objeto la cual contiene las sentencias con la información de los campos de la tabla para realizar la modificación de los datos (ver Figura 35).

Eliminación de datos a la base

```
private void EliminarActionPerformed(java.awt.event.ActionEvent evt) {  
    if(eliminarTipo != null)  
    {  
        //admin.VerificarEliminar("Activos", "id_tipo", ""+eliminarTipo.getIdTipo());  
        try  
        {  
            admin.Eliminar(TipoActivo.class, eliminarTipo.getIdTipo());  
            JOptionPane.showMessageDialog(this, "Tipo de activo eliminado", "Eliminación de datos", 1);  
            eliminarTipo = null;  
        }  
        catch(Exception e){  
            System.out.println("Error de Eliminación " + e);  
            JOptionPane.showMessageDialog(this, "Tipo utilizado en activos fijos, no es posible eliminar", "Eliminación de dato", 0);  
        }  
    }  
    else  
    {  
        JOptionPane.showMessageDialog(this, "No se ha seleccionado un tipo de activo para eliminar", "Error al eliminar", 0);  
    }  
    this.cargarTabla();  
}
```

Figura 35. Eliminación de datos a la base

Imagen elaborada por: Carlos Álvarez

- v) El método **Eliminar** de la Clase Administrador, hace un llamado a la clase **EntityManager** del paquete de persistencia, la cual nos permite realizar la operación de eliminación sobre la base de datos, el ciclo de vida de la instancia consiste en realizar un llamado al método begin() que inicializa la transacción, luego continua con la búsqueda del objeto persistente mediante el método find(), posteriormente se aplica el método remove() para realizar la eliminación del objeto y termina con el método commit() el cual confirma la eliminación de los datos (ver Figura 36).

Método eliminar

```
public void Eliminar(Class clase, Object pk) {  
    EntityManager em = getEMF().createEntityManager();  
    try {  
        em.getTransaction().begin();  
        Object obj = em.find(clase, pk);  
        em.remove(obj);  
        em.getTransaction().commit();  
    } catch (Exception e) {  
        em.getTransaction().rollback();  
        throw new PersistenceException(e);  
    } finally {  
        em.close();  
    }  
}
```

Figura 36. Método eliminar

Elaborado por: C. Álvarez y J. Navas

4.3.3. Métodos de la clase EntityManager

Los métodos de la clase Entity Manager puede realizar operaciones (Create, Read, Update, Delete) sobre un conjunto de objetos persistentes. Proporciona métodos para gestionar el ciclo de vida de las instancias entidad y para crear instancias Query, a continuación se muestra el código de la clase el cual es generado automáticamente por la persistencia:

```
package javax.persistence;
import java.util.List;
import java.util.Map;
import javax.persistence.criteria.CriteriaBuilder;
import javax.persistence.criteria.CriteriaDelete;
import javax.persistence.criteria.CriteriaQuery;
import javax.persistence.criteria.CriteriaUpdate;
import javax.persistence.metamodel.Metamodel;

public interface EntityManager {

    public void persist(Object o);

    public <T extends Object> T merge(T t);

    public void remove(Object o);

    public <T extends Object> T find(Class<T> type, Object o);

    public <T extends Object> T find(Class<T> type, Object o, Map<String,
Object> map);

    public <T extends Object> T find(Class<T> type, Object o,
LockModeType lmt);

    public <T extends Object> T find(Class<T> type, Object o,
LockModeType lmt, Map<String, Object> map);

    public <T extends Object> T getReference(Class<T> type, Object o);

    public void flush();

    public void setFlushMode(FlushModeType fmt);

    public FlushModeType getFlushMode();

    public void lock(Object o, LockModeType lmt);
```

```

    public void lock(Object o, LockModeType lmt, Map<String, Object>
map);

    public void refresh(Object o);

    public void refresh(Object o, Map<String, Object> map);

    public void refresh(Object o, LockModeType lmt);

    public void refresh(Object o, LockModeType lmt, Map<String, Object>
map);

    public void clear();

    public void detach(Object o);

    public boolean contains(Object o);

    public LockModeType getLockMode(Object o);

    public void setProperty(String string, Object o);

    public Map<String, Object> getProperties();

    public Query createQuery(String string);

    public <T extends Object> TypedQuery<T>
createQuery(CriteriaQuery<T> cq);

    public Query createQuery(CriteriaUpdate cu);

    public Query createQuery(CriteriaDelete cd);

    public <T extends Object> TypedQuery<T> createQuery(String string,
Class<T> type);

    public Query createNamedQuery(String string);

    public <T extends Object> TypedQuery<T> createNamedQuery(String
string, Class<T> type);

    public Query createNativeQuery(String string);

    public Query createNativeQuery(String string, Class type);

    public Query createNativeQuery(String string, String string1);

    public StoredProcedureQuery createNamedStoredProcedureQuery(String
string);

```

```

    public StoredProcedureQuery createStoredProcedureQuery(String string);

    public StoredProcedureQuery createStoredProcedureQuery(String string,
Class[] types);

    public StoredProcedureQuery createStoredProcedureQuery(String string,
String[] strings);

    public void joinTransaction();

    public boolean isJoinedToTransaction();

    public <T extends Object> T unwrap(Class<T> type);

    public Object getDelegate();

    public void close();

    public boolean isOpen();

    public EntityTransaction getTransaction();

    public EntityManagerFactory getEntityManagerFactory();

    public CriteriaBuilder getCriteriaBuilder();

    public Metamodel getMetamodel();

    public <T extends Object> EntityGraph<T> createEntityGraph(Class<T>
type);

    public EntityGraph<?> createEntityGraph(String string);

    public EntityGraph<?> getEntityGraph(String string);

    public <T extends Object> List<EntityGraph<? super T>>
getEntityGraphs(Class<T> type);
}

```

4.3.4. Métodos de la clase EntityManagerFactory

Los métodos de la EntityManagerFactory nos ayudan a crear objetos de EntityManager, proporciona métodos para la creación de instancias, a continuación se muestra el código de la clase el cual es generado automáticamente por la persistencia

```
package javax.persistence;
```

```

import java.util.Map;
import javax.persistence.criteria.CriteriaBuilder;
import javax.persistence.metamodel.Metamodel;

public interface EntityManagerFactory {

    public EntityManager createEntityManager();

    public EntityManager createEntityManager(Map map);

    public EntityManager createEntityManager(SynchronizationType st);

    public EntityManager createEntityManager(SynchronizationType st,
Map map);

    public CriteriaBuilder getCriteriaBuilder();

    public Metamodel getMetamodel();

    public boolean isOpen();

    public void close();

    public Map<String, Object> getProperties();

    public Cache getCache();

    public PersistenceUnitUtil getPersistenceUnitUtil();

    public void addNamedQuery(String string, Query query);

    public <T extends Object> T unwrap(Class<T> type);

    public <T extends Object> void addNamedEntityGraph(String string,
EntityGraph<T> eg);
}

```

4.3.5. Métodos de la clase EntityTransaction

Cada instancia de Entity Manager tiene una relación de uno a uno con una instancia de EntityTransaction. Maneja el concepto de todos o ninguno para mantener la integridad de los datos.

```

package javax.persistence;

public interface EntityTransaction {

```

```

public void begin();

public void commit();

public void rollback();

public void setRollbackOnly();

public boolean getRollbackOnly();

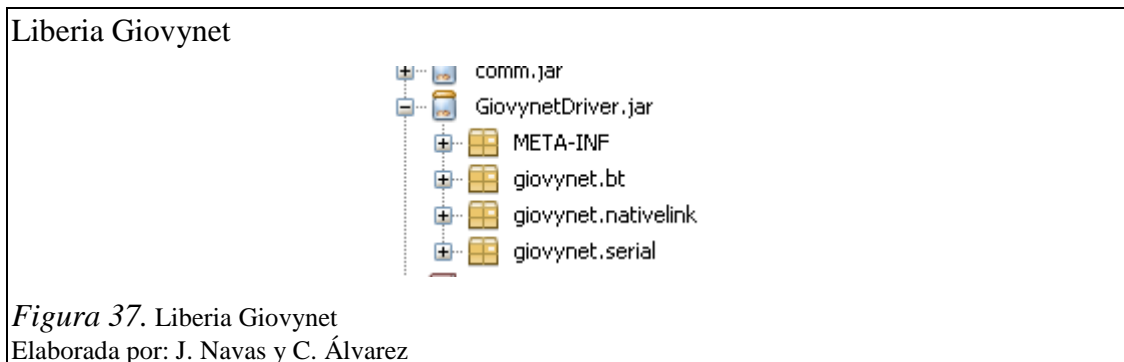
public boolean isActive();
}

```

4.3.6. Comunicación entre la aplicación y el lector RFID

Para realizar la comunicación con el software de inventario y los lectores RFID, se procedió a importar la librería que permite la lectura de un puerto serial que posee el lector hacia el puerto USB del ordenador mediante un cable USB-SERIAL, a continuación se describe los pasos para realizar la conectividad con el dispositivo:

- i) Se utiliza la Librería GiovynetDriver.jar la cual permite importar paquetes para utilizarlos en la comunicación con los dispositivos (ver Figura 37).



- ii) La siguiente clase permite establecer el puerto por el cual se realizara la comunicación (ver Figura 38).

Clase puerto de comunicación

```
public RFID() {  
    initComponents();  
  
    try{  
        Conect("COM4");  
    }catch(Exception e){  
  
    }  
}
```

Figura 38. Clase puerto de comunicación
Elaborada por: J. Navas y C. Álvarez

- iii) Se ha creado una clase **Conect** que realiza la apertura del puerto antes seleccionado en la cual se establecen los parámetros de conexión que utiliza el puerto (ver Figura 39)

Clase conect

```
public void Conect(String COM) throws Exception {  
    portId = CommPortIdentifier.getPortIdentifier(COM);  
  
    try {  
        serialPort = (SerialPort) portId.open("ComControl",2000);  
        System.out.println("PUERTO : "+COM+" ABIERTO Y LISTO");  
    } catch (PortInUseException e) { System.out.println("PUERTO OCUPADO"); }  
    try {  
        serialPort.setSerialPortParams(9600,  
                                         SerialPort.DATABITS_8,  
                                         SerialPort.STOPBITS_1,  
                                         SerialPort.PARITY_NONE);  
  
        serialPort.setDTR(false);  
        serialPort.setRTS(false);  
    } catch (UnsupportedCommOperationException e) { System.err.println("Wrong Parameters"); }  
    try {  
        inputStream = serialPort.getInputStream();  
    } catch (IOException ex) { System.err.append("ERROR DE LECTURA"); }  
    try {  
        outputStream = serialPort.getOutputStream();  
    } catch (IOException ex) { System.err.append("ERROR DE ESCRITURA"); }  
    serialPort.notifyOnDataAvailable(true);  
    readThread.start();  
}
```

Figura 39. Clase conect
Elaborado por: J. Navas y C. Álvarez

- iv) La siguiente clase **Read** como su nombre lo indica realiza la lectura del Tag una vez que pasa por el lector previamente conectado (ver Figura 40).

Clase read

```
public void Read() {  
    try {  
        int dato=inputStream.read();  
        dato=dato-48;  
        dat=dat+dato;  
        //cadena=dato+cadena;  
        jTextField1.setText(dat);  
        System.out.println("dato recibido " + dato);  
    } catch (IOException ex) { System.err.println("NO SE HA LEIDO"); }  
}
```

Figura 40. Clase read

Elaborada por: J. Navas y C. Álvarez

- v) Se configura un timer para que cada 500 milisegundos el lector realice una lectura constante y realice un barrido de información (ver Figura 41).

Timer

```
private void formWindowOpened(java.awt.event.WindowEvent evt) {  
    // TODO add your handling code here:  
    // TODO add your handling code here:  
    objTimer=new Timer(500,new ClaseTimer());  
  
    //Read();  
}
```

Figura 41. Timer

Elaborado por: J. Navas y C. Álvarez

4.4. Implementación del sistema

Las herramientas utilizadas para la implementación del sistema fueron:

- Servidor de base de datos HP ProLiant DL380 Generation 7, procesador Intel® Xeon® E5640 (2.7GHz/4-core/12MB/80W, DDR3-1066, HT, Turbo 1/1/2/2), sistema operativo Windows Server 2008 R2 Standard 64-bit, 16 GB en RAM, 465 GB (ver Figura 42).

Servidor HP Proliant DL380 G7



Figura 42. Servidor HP Proliant DL380 G7

Fuente: HP

- Computadores HP Compaq 6000 NV503UT Pro Desktop PC - Intel Core 2 Quad Q94000 2.6GHz, 4GB DDR3 500GB HDD, DVDRW, Windows XP Service Pack 3 (Equipos que utilizan el Personal de Seguridad Física) (ver Figura 43).

Computador HP Compaq 6000 Pro



Figura 43. Computador HP Compaq 6000 Pro

Fuente: sitio HP

Se utiliza el sistema operativo de la maquina debido a que el área de seguridad física posee un sistema de control de vigilancia de cámaras el cual corre bajo el mismo.

- Laptop Dell Latitude E6440 - Intel® Core™ i7-4600M CPU @ 2.90GHz (4 CPUs), ~2.9GHz, 4GB DDR3 500GB HDD, Windows 8.1 Enterprise 64-bit (Equipos utilizados por el administrador del sistema y la persona de Activos Fijos) (ver Figura 44).

Laptop Dell Latitude E6440



Figura 44. Laptop Dell Latitude E6440

Fuente: Dell

- Lectores RFID (adquiridos y ensamblado en tienda electrónica) (ver Figura 45), cables de conexión USB-Serial (conexión entre dispositivo y PC) (ver Figura 46).

Lector RFID

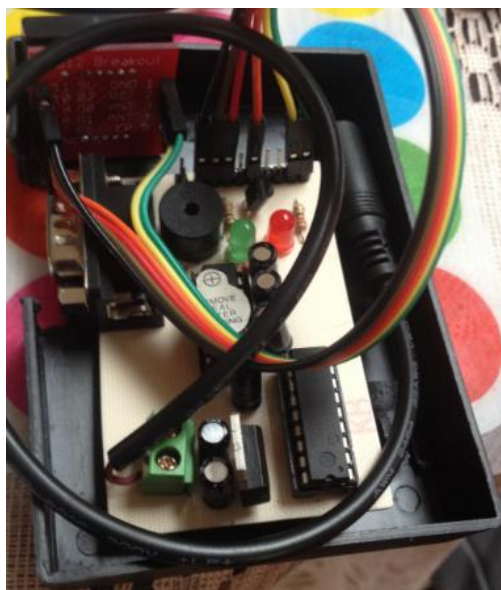


Figura 45. Lector RFID

Elaborado por: C. Álvarez y J. Navas

Cable USB serial



Figura 46. Cable USB serial
Elaborado por: C. Álvarez y J. Navas

- Etiquetas RFID (adquiridas en tienda electrónica) ver Figura 47.



Figura 47. Etiquetas RFID
Imagen tomada por Carlos Álvarez
(Fecha de actualización 2015)

- Se utilizó el Ide de Java Netbeans para el desarrollo del sistema (ver Figura 48).

Logo Netbeans



Figura 48. Logo Netbeans
Fuente: Netbeans

- SQL Server 2008 R2 motor de base de datos que alberga la información del sistema (ver Figura 49).

Logo SQL Server 2008 R2



Figura 49. Logo SQL Server 2008 R2

Fuente: MSDN

A continuación se describe el proceso de instalación de la base de datos del sistema de control de activos en el servidor:

- i) Se Abrirá automáticamente la instalación como en la imagen (ver Figura 50).

Inicio de instalación SQL Server

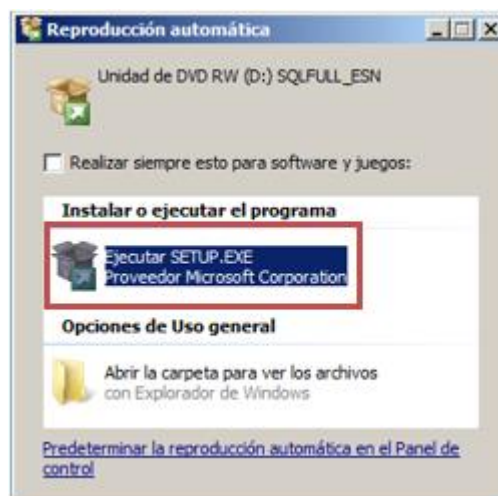


Figura 50. Inicio de instalación SQL Server

Elaborado por: J. Navas y C. Álvarez

- ii) Se escogerá la opción instalar y la opción nueva instalación o agregar características a una instalación existente (ver Figura 51).

Nueva instalación SQL Server



Figura 51. Nueva instalación SQL Server

Elaborador por: J. Navas y C. Álvarez

- iii) En esta ventana se da en la opción siguiente escribiendo la clave de activación del producto (ver Figura 52).

Clave de producto SQL Server 2008



Figura 52. Clave de producto SQL Server 2008

Elaborado: por J. Navas y C. Álvarez

- iv) Aceptar los términos de la licencia y dar siguiente (ver Figura 53).

Términos de licencia SQL Server 2008

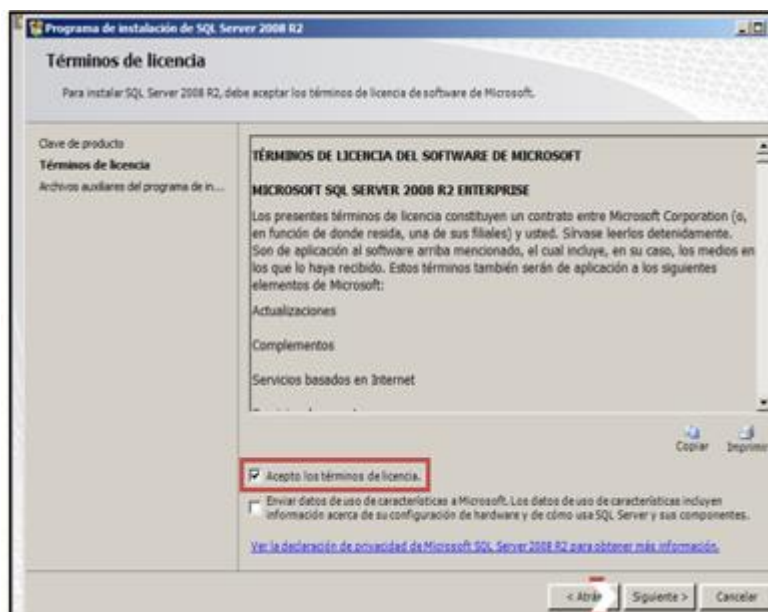


Figura 53. Términos de licencia SQL Server 2008

Elaborado por: J. Navas y C. Álvarez

- v) Al terminar la instalación se abrirá la ventana de reglas auxiliares del programa de instalación (ver Figura 54).

Reglas auxiliares SQL Server 2008

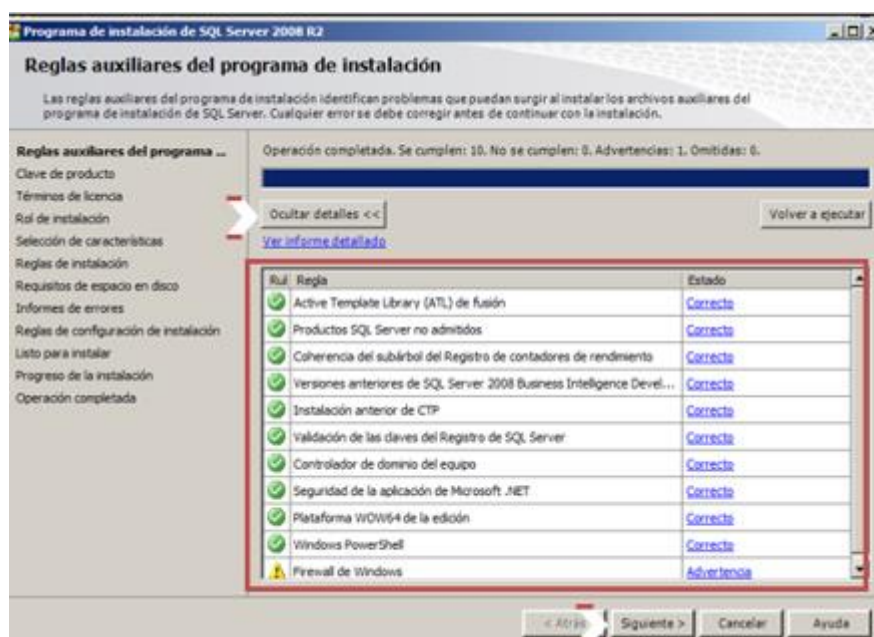


Figura 54. Reglas auxiliares SQL Server 2008

Elaborado por J. Navas y C. Álvarez

Lo más recomendable es hacer clic Usar la misma cuenta para todos los servidores de SQL Server.

- vi) Seleccionar AUTHORITY\SYSTEM y Aceptar. Se tendrá como resultado al finalizar esta ventana y dar siguiente (ver Figura 55).

Configuración servidor SQL Server 2008

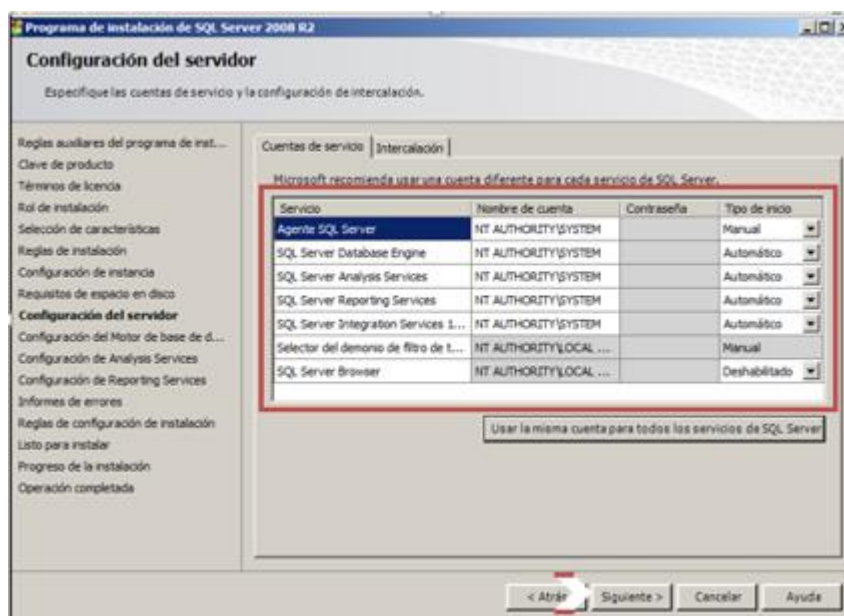


Figura 55. Configuración servidor SQL Server 2008

Elaborado por: J. Navas y C. Álvarez

- vii) Finalmente te mostrara esta pantalla con este mensaje (ver Figura 56).

Finalización de instalación SQL Server 2008

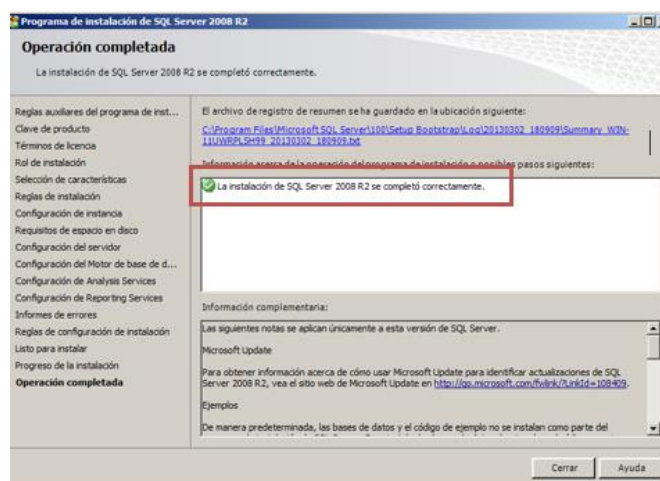


Figura 56. Finalización de instalación SQL Server 2008

Elaborado por: J. Navas y C. Álvarez

viii) Comprobar que la instalación es correcta (ver Figura 57).

Comprobación instalación SQL Server 2008

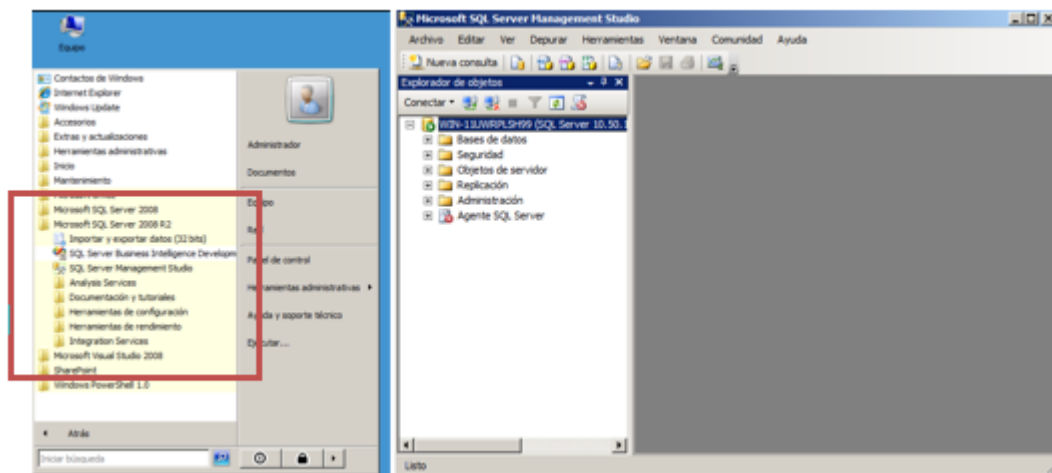


Figura 57. Comprobación instalación SQL Server 2008

Elaborado por: J. Navas y C. Álvarez

ix) La base de datos lleva el nombre de ActivoSistemas, la cual contendrá la información del sistema (ver Figura 58).

Base de datos

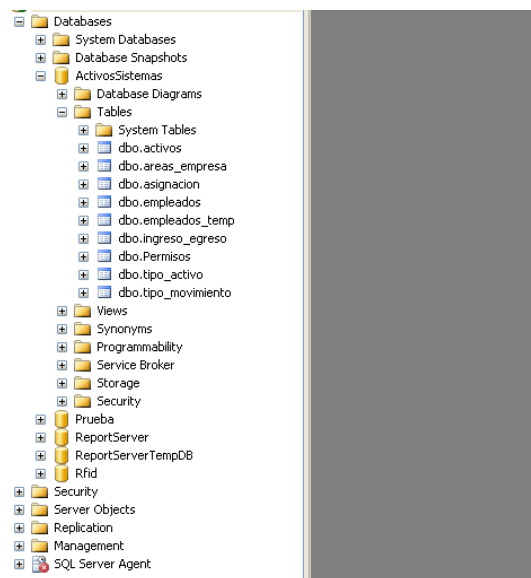


Figura 58. Base de datos

Elaborado por: J. Navas y C. Álvarez

4.5. Administración y manejo del sistema

4.5.1. Pantalla inicio de sesión

Ventana de acceso (ver Figura 59) para el usuario de acuerdo a los roles definidos.

Pantalla principal de inicio de sesión



Figura 59. Pantalla principal de inicio de sesión
Elaborado por: J. Navas y C. Álvarez

El sistema está compuesto actualmente por tres perfiles los cuales fueron determinados en el análisis del proyecto, a continuación se mostrará la utilización del sistema por cada uno de los perfiles:

4.5.2. Perfil administrador

Pantalla principal del perfil administrador (ver Figura 60).

Pantalla principal administrador

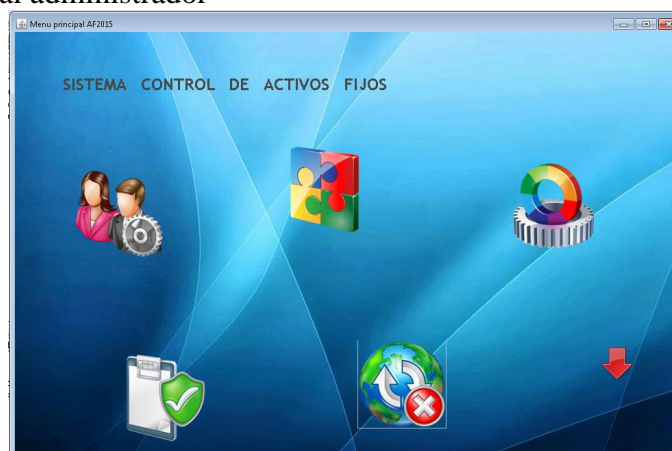


Figura 60. Pantalla principal administrador
Elaborado por: J. Navas y C. Álvarez

i) Pantalla de ingreso de empleados (ver Figura 61)

Pantalla ingreso de empleados perfil administrador

DETALLE GENERAL EMPLEADOS AF - 2015

Cédula	Nombres	Area	Nombre
0501804678	Claudia Corral	1	Finanzas
0502227168	Diana Bustos	5	Ingeniería
0502658461	David Montaluisa	5	Ingeniería
0603609371	Juan Vallejo	5	Ingeniería
0603885104	José Navas	2	Sistemas
0800407124	Cesar Gordillo	1	Finanzas
0801856964	Cesar Madrid	1	Finanzas
1500434947	Betty Ledesma	3	Operaciones
1705252904	Angel Ushirfa	3	Operaciones
1706862628	Jose Sevilla	1	Finanzas
1710891381	Jenny Simbaria	5	Ingeniería
1713037941	Giovanny Maldonado	1	Finanzas
1713539615	Mario Jurado	1	Finanzas
1713706164	Alvaro Peñaherrera	5	Ingeniería
1714365532	Claudia Baquero	1	Finanzas
1715830616	Alejandro Salcedo	5	Ingeniería
1716562010	Alejandra Mosquera	1	Finanzas
1716786148	Alejandra Mosquera	1	Finanzas
1718829359	Arceli Cruz	4	Compras
1719384677	Miguel Hidalgo	5	Ingeniería
1719937276	Manuel Muñoz	1	Finanzas
1801938596	Jorge Aldaz	5	Ingeniería
1803749934	Alex Garcia	5	Ingeniería

Figura 61. Pantalla ingreso de empleados perfil administrador

Elaborado por: J. Navas y C. Álvarez

- ii) Dentro de la pantalla de ingreso de empleados se encuentra un submenú en el cual nos permite crear nuevas áreas, también realizar la asignación de un equipo ya asignado a un usuario autorizado (ver Figura 62).

Pantalla de submenú empleados

submenú empleados

ID	AREA
1	Finanzas
2	Sistemas
3	Operaciones
4	Compras
5	Ingeniería
6	Servicios Generales
7	Auditoria
8	Ofertas
9	Comunicación
10	Gerencia General
11	Legal

Figura 62. Pantalla de submenú empleados

Elaborado por: J. Navas y C. Álvarez

- iii) En la pantalla de áreas de la empresa (ver Figura 63) se podrá adicionar nuevas áreas en el caso que se requiere, actualizarlas o eliminarlas.

Pantalla áreas de la empresa

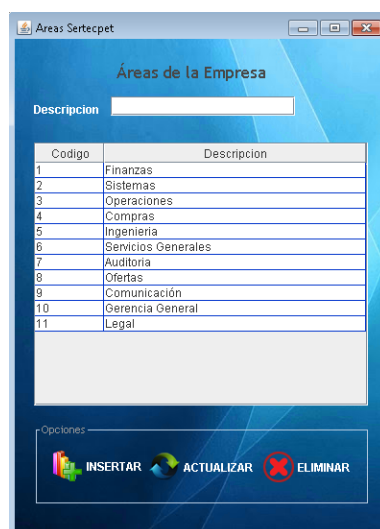


Figura 63. Pantalla áreas de la empresa
Elaborado por: J. Navas y C. Álvarez

- iv) Asignación de activos entre empleados (ver Figura 64)

Pantalla asignación entre usuarios



Figura 64. Pantalla asignación entre usuarios
Elaborado por: J. Navas y C. Álvarez

v) Pantalla de ingreso de activos informáticos (ver Figura 65)

Pantalla ingreso de activos informáticos

Id Activo	Estado	Tipo	Nombre	Características	Serie	Rfid	Fecha
3	✓	1	Laptop	Latitude E6440 Intel(R) Core(TM) i7-3...	9ULRWZ1	511893288696	05/03/2015
4	✓	1	Laptop	Pavilion 14 Core i5	5CD3342P7W	511833856907	05/03/2015
5	✓	1	Laptop	Portege Z930 Intel(R) Core(TM) i5-33...	2D035049H	510633834607	05/03/2015
6	✓	1	Laptop	Portege Z930 Intel(R) Core(TM) i5-33...	3D052150H	511635854507	05/03/2015
7	✓	13	CPU	HP ProBook 4540s Intel(R) Core(TM) i...	8KLN8745	511789428537	05/03/2015
8	✓	13	CPU	ProDesk 600 G1 SFF Intel(R) Core(TM) i...	83406065	511587192345	05/03/2015
9	✓	13	CPU	OptiPlex 7010 Intel(R) Core(TM) i7-33...	3KL5RW1	511923410349	05/03/2015
10	✓	10	Workstation	EliteBook 8470p Intel(R) Core(TM) i7...	CNU329C19P	511096501278	05/03/2015
11	✓	1	Laptop	HP ProBook 4540s Intel(R) Core(TM) i...	2CE2402050	511678340645	05/03/2015
12	✓	1	Laptop	HP ProBook 4440s Intel(R) Core(TM) i...	DL19QW1	511234529460	05/03/2015
13	✓	2	Monitor	Samsung 23" Led E2320	235GH56WQR	511892374612	05/03/2015
14	✓	2	Monitor	Samsung 23" Led E2320	235GH56QRT	511789345261	05/03/2015
15	✓	2	Monitor	Samsung 18" Led E1850	134RTY54HG378	511009893623	05/03/2015
16	✓	2	Monitor	Samsung 18" Led E1850	134RTY54HG768	511005623756	05/03/2015
17	✓	7	Access Point	Cisco E3430 56HZ	GHRT2340T	511673821566	05/03/2015
18	✓	7	Access Point	Cisco E3440 56HZ	567HRT67T	511127529150	05/03/2015
19	✓	11	Servidor	HP Proliant G8 32GB RAM	2345GRT62YU4	511722345179	05/03/2015

Figura 65. Pantalla ingreso de activos informáticos

Elaborado por: J. Navas y C. Álvarez

vi) Dentro de la Pantalla de ingreso de activos se encuentra un submenú en el cual nos permite crear tipos de activos electrónicos (ver Figura 66).

Pantalla submenú activos

Figura 66. Pantalla submenú activos

Elaborado por: J. Navas y C. Álvarez

- vii) En la pantalla de ingreso de activos se podrá adicionar un nuevo tipo de activo si requiere el caso (ver Figura 67).

Pantalla submenú activos

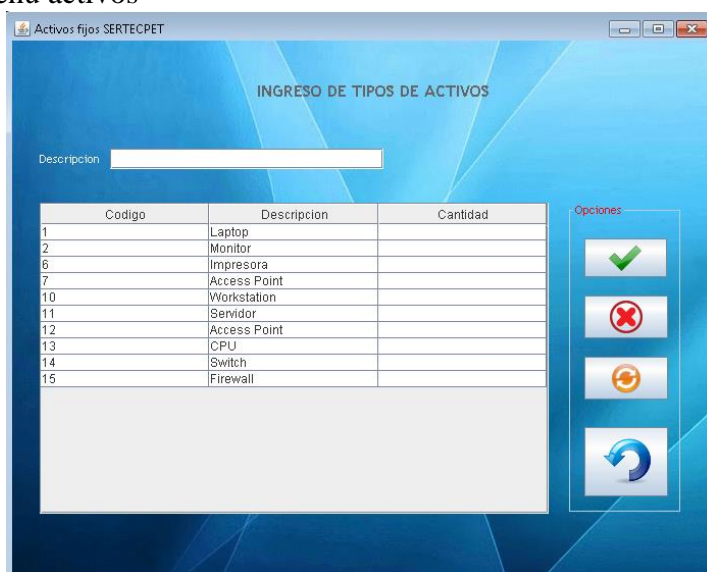


Figura 67. Pantalla submenú activos
Elaborado por: J. Navas y C. Álvarez

- viii) En la pantalla de asignación el activo informático será atado a un empleado (ver Figura 68).

Pantalla asignación de activos



Figura 68. Pantalla asignación de activos
Elaborado por: J. Navas y C. Álvarez

- ix) En el módulo de configuración se puede crear perfiles según la necesidad amerite, o actualizar los ya creados, actualmente se establecieron 4 perfiles (ver Figura 69).

Pantalla asignación de activos

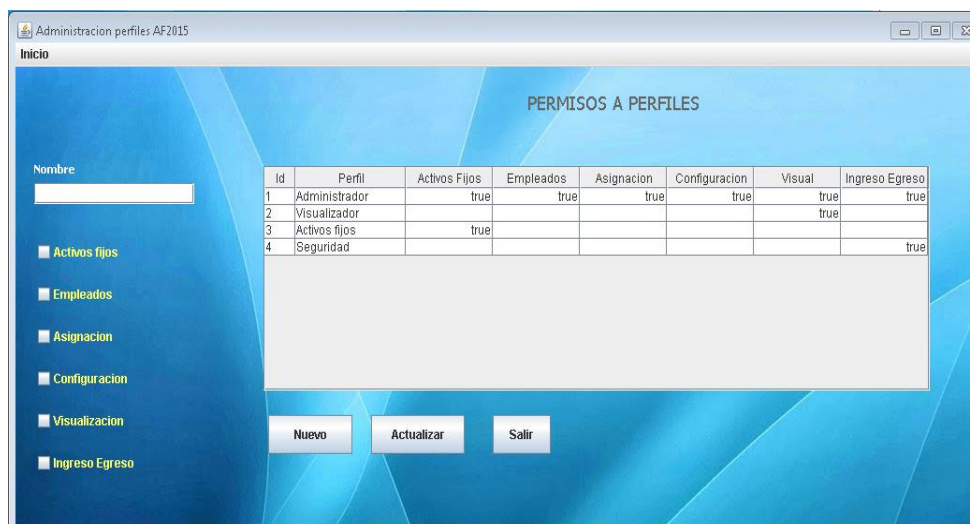


Figura 69. Pantalla asignación de activos

Elaborado por: J. Navas y C. Álvarez

- x) Dentro de la pantalla de perfiles se encuentra un submenú en el cual nos permite crear nuevos usuarios que accederán al sistema (ver Figura 70).

Pantalla submenú perfiles

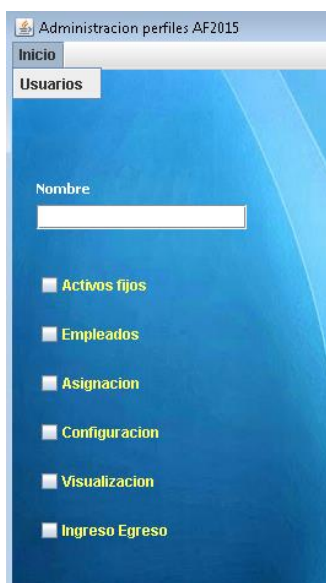


Figura 70. Pantalla submenú perfiles

Elaborado por: J. Navas y C. Álvarez

xi) Pantalla de creación de usuarios para acceso al sistema (ver Figura 71).

Pantalla creación de usuario

id	Perfil
1	Administrador
2	Visualizador
3	Activos fijos
4	Seguridad

Usuario	Nombres	id Perfil	Nombre
calvarez	Carlos Alvarez	1	Administrador
dtorres	Daniel David Torres	2	Visualizador
dvallejo	Daniel Vallejo	1	Administrador
jnavas	Jose Navas	1	Administrador
lerazo	Luis Erazo	2	Visualizador
mvasconez	Milton Vasconez	3	Activos fijos
nmosquera	Norman Mosquera	4	Seguridad

Figura 71. Pantalla creación de usuario

Elaborado por: J. Navas y C. Álvarez

xii) En todos los módulos del sistema existe una opción en el submenú que permite imprimir un reporte previamente diseñado por la herramienta iReport, la cual se acopla a nuestro sistema (ver Figura 72).

Pantalla submenú imprimir

Figura 72. Pantalla submenú imprimir

Elaborado por: J. Navas y C. Álvarez

xiii) La siguiente pantalla muestra el reporte que se puede obtener por la opción de imprimir (ver Figura 73), cabe recalcar que una vez ejecutado el reporte este puede ser impreso físicamente o mediante una impresora pdf.

Pantalla impresión de reporte



USUARIO	CONTRASEÑA	DATOS COMPLETOS	PERFIL
calvarez	carlos	Carlos Alvarez	Administrador
dtorres	daniel	Daniel David Torres	Visualizador
dvallejo	daniel	Daniel Vallejo	Administrador
jnavas	jose	Jose Navas	Administrador
lerazo	luis	Luis Erazo	Visualizador
mvasconez	milton	Milton Vasconez	Activos fijos
nmosquera	norman	Norman Mosquera	Seguridad
pperez	pepe	Pepito	Visualizador

Figura 73. Pantalla impresión de reporte
Elaborado por: J. Navas y C. Álvarez

xiv) Diagrama de procesos del perfil administrador (ver Figura 74).

Diagrama de procesos perfil administrador

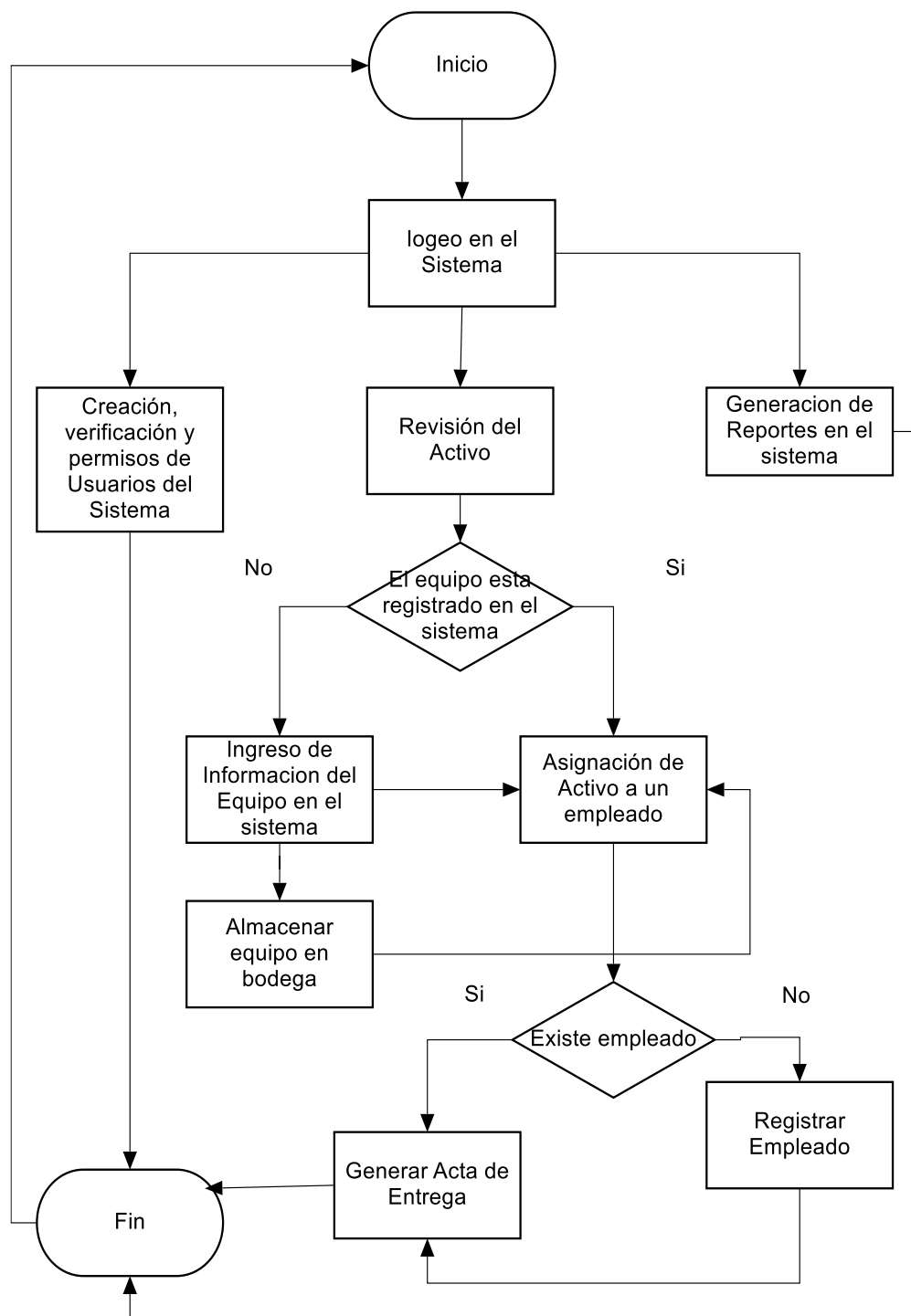


Figura 74. Diagrama de procesos perfil administrador
Elaborado por: J. Navas y C. Álvarez

4.5.3. Perfil visualizador

i) Pantalla principal del perfil visualizador (ver Figura 75).

Pantalla principal perfil visualizador



Figura 75. Pantalla principal perfil visualizador
Elaborado por: J. Navas y C. Álvarez

ii) Pantalla de egreso de empleados (ver Figura 76)

Pantalla ingreso de empleados perfil visualizador

Cédula	Nombres	Area	Nombre
0501804670	Claudia Corral	1	Finanzas
0502227188	Diana Bustos	5	Ingeniería
0502658461	David Montalusa	5	Ingeniería
0603609371	Juan Vallejo	5	Ingeniería
0603895104	José Navas	2	Sistemas
0800407124	Cesar Gordillo	1	Finanzas
0801856964	Cesar Madrid	1	Finanzas
1500434947	Betty Ledesma	3	Operaciones
1705262904	Angel Ushiffa	3	Operaciones
1706862628	José Sevilla	1	Finanzas
1710891381	Jenny Simbaña	5	Ingeniería
1713037941	Giovanny Maldonado	1	Finanzas
1713539815	Mario Jurado	1	Finanzas
1713706184	Alvaro Peñaherrera	5	Ingeniería
1714355532	Claudia Baquero	1	Finanzas
1715830816	Alejandro Salcedo	5	Ingeniería
1716562010	Alejandra Mosquera	1	Finanzas
1716786148	Alejandra Mosquera	1	Finanzas
1718829359	Araceli Cruz	4	Compras
1719384677	Miguel Hidalgo	5	Ingeniería
1719937276	Manuel Muñoz	1	Finanzas
1801938596	Jorge Aldaz	5	Ingeniería
1803749934	Alex Garcia	5	Ingeniería

Figura 76. Pantalla ingreso de empleados perfil visualizador
Elaborada por: C. Álvarez

iii) Pantalla submenú de áreas existentes en la empresa (ver Figura 77)

Pantalla áreas de la empresa perfil visualizador

Áreas de la Empresa

Descripcion

Codigo	Descripcion
1	Finanzas
2	Sistemas
3	Operaciones
4	Compras
5	Ingenieria
6	Servicios Generales
7	Auditoria
8	Ofertas
9	Comunicación
10	Gerencia General
11	Legal

Opciones

INSERTAR
 ACTUALIZAR
 ELIMINAR

Figura 77. Pantalla áreas de la empresa perfil visualizador
Elaborado por: J. Navas y C. Álvarez

iv) Pantalla de submenú asignación de activos entre empleados (ver Figura 78)

Pantalla submenú asignación entre empleados perfil visualizador

ASIGNACION DE EMPLEADOS

Archivo

ASIGNACION EMPLEADOS AUTORIZADOS

EMPLEADOS

Cedula	Nombres
0501804678	Claudia Corral
0502227168	Diana Bustos
0502658461	David Montaluisa
0603609371	Juan Vallejo
0603885104	José Navas
0800407124	Cesar Gordillo
0801856964	Cesar Madrid
1500434947	Betty Ledesma
1705252904	Angel Ushifia
1708862628	Jose Sevilla
1710891381	Jenny Simbaria
1713037941	Giovanny Maldonado
1713539615	Mario Jurado

ASIGNAR

Cedula	Nombres
0501804678	Claudia Corral
0502227168	Diana Bustos
0502658461	David Montaluisa
0603609371	Juan Vallejo
0603885104	José Navas
0800407124	Cesar Gordillo
0801856964	Cesar Madrid
1500434947	Betty Ledesma
1705252904	Angel Ushifia
1708862628	Jose Sevilla
1710891381	Jenny Simbaria
1713037941	Giovanny Maldonado
1713539615	Mario Jurado

Id Asig	Empleado	Asignado	Area
1	Manuel Muñoz	Alejandra Mosquera	Finanzas
2	Juan Vallejo	Diana Bustos	Ingenieria
3	Cesar Madrid	Giovanny Maldonado	Finanzas
4	Claudia Baquero	Claudia Corral	Finanzas
5	Araceli Cruz	Jose Sevilla	Compras
6	Alejandro Salcedo	Alvaro Peraherrera	Ingenieria
7	Betty Ledesma	Miguel Hidalgo	Operaciones
8	Alex Garcia	Jorge Aldaz	Ingenieria
9	Juan Vallejo	Jose Sevilla	Ingenieria

Figura 78. Pantalla submenú asignación entre empleados perfil visualizador
Elaborado por: J. Navas y C. Álvarez

v) Pantalla de activos informáticos (ver Figura 79)

Pantalla de activos informáticos perfil visualizador

Id Activo	Estado	Tipo	Nombre	Características	Serie	Rfid	Fecha
1		1	Laptop	Dell Latitude E6440 Core i7 8 RAM	ERF4567Ri	512456314562	05/03/2015
2	✓	1	Laptop	Latitude E6440 Intel(R) Core(TM) i7-	8TXRWZ1	543563287896	05/03/2015
3	✓	1	Laptop	Latitude E6440 Intel(R) Core(TM) i7-	QOLRWZ1	511893208686	05/03/2015
4	✓	1	Laptop	Pavilion 14 Core i5	SCD3342P7W	511833856807	05/03/2015
5	✓	1	Laptop	Portege Z930 Intel(R) Core(TM) i5-33	2D035049H	510633834607	05/03/2015
6	✓	1	Laptop	Portege Z930 Intel(R) Core(TM) i5-33	3D052150H	511635854507	05/03/2015
7	✓	13	CPU	HP ProBook 4540s Intel(R) Core(TM) i5-	8KLN8745	511789428527	05/03/2015
8	✓	13	CPU	ProDesk 600 G1 SFF Intel(R) Core(TM)	8340685	511587192345	05/03/2015
9	✓	13	CPU	OptiPlex 7010 Intel(R) Core(TM) i7-33	3KL5RW1	511923410349	05/03/2015
10	✓	10	Workstation	EliteBook 8470p Intel(R) Core(TM) i7-	CNU329C19P	511096501278	05/03/2015
11	✓	1	Laptop	HP ProBook 4540s Intel(R) Core(TM)	2CE2402050	511678340845	05/03/2015
12	✓	1	Laptop	HP ProBook 4440s Intel(R) Core(TM)	DL19QW1	511234528460	05/03/2015
13	✓	2	Monitor	Samsung 23" Led E2320	2350H56WQR	511892374612	05/03/2015
14	✓	2	Monitor	Samsung 23" Led E2320	2350H56GORT	511789345261	05/03/2015
15	✓	2	Monitor	Samsung 18" Led E1850	134RTY54HG378	51100893623	05/03/2015
16	✓	2	Monitor	Samsung 18" Led E1850	134RTY54HG768	511005623756	05/03/2015
17	✓	17	Access Point	Cisco F3430 50H7	ICMRT7740IT	511673871566	05/03/2015

Figura 79. Pantalla de activos informáticos perfil visualizador

Elaborado por: J. Navas y C. Álvarez

vi) Pantalla de submenú tipos de activos informáticos (ver Figura 80).

Pantalla submenú tipos de activos informáticos perfil visualizador

Codigo	Descripcion	Cantidad
1	Laptop	
2	Monitor	
6	Impresora	
7	Access Point	
10	Workstation	
11	Servidor	
12	Access Point	
13	CPU	
14	Switch	
15	Firewall	

Figura 80. Pantalla submenú tipos de activos informáticos perfil visualizador

Elaborado por: J. Navas y C. Álvarez

vii) Pantalla de asignación el activo informático a un empleado (ver Figura 81).

Pantalla asignación de activos informáticos perfil visualizador

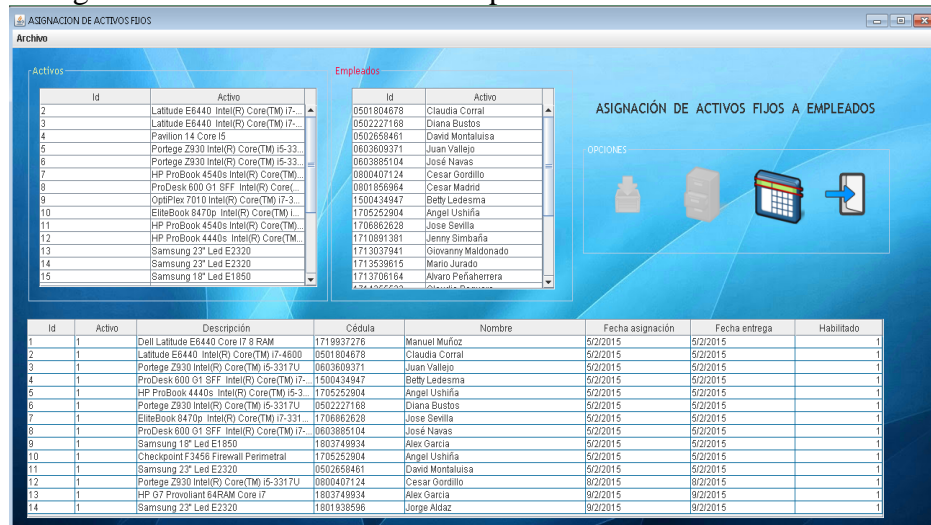


Figura 81. Pantalla asignación de activos informáticos perfil visualizador

Elaborado por: J. Navas y C. Álvarez

viii) Pantalla de configuración, acceso a perfiles de usuario del sistema (ver Figura 82)

Pantalla de configuración perfil visualizador



Figura 82. Pantalla de configuración perfil visualizador

Elaborado por: J. Navas y C. Álvarez

ix) La siguiente pantalla muestra el reporte que se puede obtener por la opción de imprimir, los cuales son ubicados dentro de los submenús (ver Figura 83).

Pantalla impresión de reporte perfil visualizador



ID	CEDULA	EMPLEADO	AREA EMPRESA
1	1719937276	Manuel Muñoz	Finanzas
TIPO	Laptop		
DESCRIPCION	Dell Latitude E6440 Core i7 8 RAM		
DESDE	5/2/2015		
HASTA	5/2/2015		
ID	CEDULA	EMPLEADO	AREA EMPRESA
2	0501804678	Claudia Corral	Finanzas
TIPO	Laptop		
DESCRIPCION	Latitude E6440 Intel(R) Core(TM) i7-4600		
DESDE	5/2/2015		
HASTA	5/2/2015		
ID	CEDULA	EMPLEADO	AREA EMPRESA
3	0603609371	Juan Vallejo	Ingeniería
TIPO	Laptop		
DESCRIPCION	Portege Z930 Intel(R) Core(TM) i5-3317U		
DESDE	5/2/2015		

Figura 83. Pantalla impresión de reporte perfil visualizador
Elaborado por: J. Navas y C. Álvarez

x) Diagrama de procesos del perfil visualizador (ver Figura 84).

Diagrama de procesos perfil visualizador

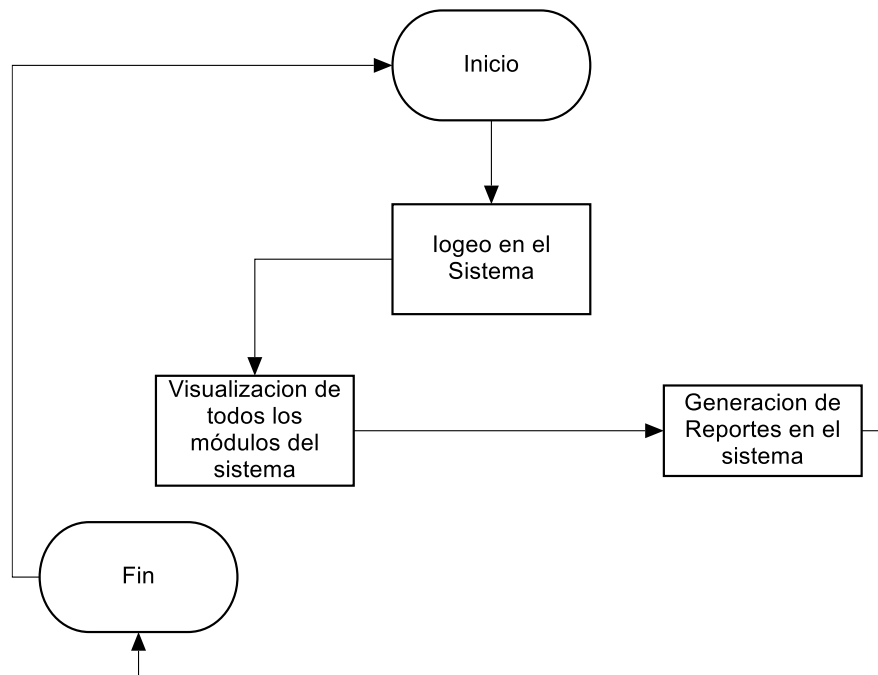


Figura 84. Diagrama de procesos perfil visualizador
Elaborado por: J. Navas y C. Álvarez

4.5.4. Perfil activos fijos

i) Pantalla principal del perfil activos fijos (ver Figura 85).

Pantalla principal perfil activos fijos



Figura 85. Pantalla principal perfil activos fijos
Elaborado por: J. Navas y C. Álvarez

ii) Pantalla de activos informáticos (ver Figura 86).

Pantalla activos informáticos perfil activos fijos

Id Activo	Estado	Tipo	Nombre	Características	Serie	Rfid	Fecha
1		1	Laptop	Dell Latitude E6440 Core i7 8 RAM	ERF4567RI	512456314562	05/03/2015
2	✓	1	Laptop	Latitude E6440 Intel(R) Core(TM) i7-...	8TXRWZ1	543563287696	05/03/2015
3	✓	1	Laptop	Latitude E6440 Intel(R) Core(TM) i7-...	GQLRWZ1	511893288696	05/03/2015
4	✓	1	Laptop	Pavilion 14 Core i5	5CD3342P7W	511833856907	05/03/2015
5	✓	1	Laptop	Portege Z930 Intel(R) Core(TM) i5-33...	2D035049H	510633834607	05/03/2015
6	✓	1	Laptop	Portege Z930 Intel(R) Core(TM) i5-33...	3D092150H	511635854507	05/03/2015
7	✓	13	CPU	HP ProBook 4540s Intel(R) Core(TM)	8KLN8745	511789428537	05/03/2015
8	✓	13	CPU	ProDesk 600 G1 SFF Intel(R) Core(T	83406065	511587192345	05/03/2015
9	✓	13	CPU	OptiPlex 7010 Intel(R) Core(TM) i7-33	3KL5RW1	511923410349	05/03/2015
10	✓	10	Workstation	EliteBook 8470p Intel(R) Core(TM) i7	CNU329C19P	511098501278	05/03/2015
11	✓	1	Laptop	HP ProBook 4540s Intel(R) Core(TM)	2CE2402050	511678340645	05/03/2015
12	✓	1	Laptop	HP ProBook 4440s Intel(R) Core(TM)	DL19QW1	511234529460	05/03/2015
13	✓	2	Monitor	Samsung 23" Led E2320	235GH56WQR	511892374612	05/03/2015
14	✓	2	Monitor	Samsung 23" Led E2320	235GH56QRT	511789345261	05/03/2015
15	✓	2	Monitor	Samsung 18" Led E1850	134RTY54HG378	511009893623	05/03/2015
16	✓	2	Monitor	Samsung 18" Led E1850	134RTY54HG768	511005623756	05/03/2015
17	✓	17	Access Point	Cisco F3430 5GHz	IGHRT2340T	511673821566	05/03/2015

Figura 86. Pantalla activos informáticos perfil activos fijos
Elaborado por: J. Navas y C. Álvarez

iii) Pantalla de submenú tipos de activos informáticos (ver Figura 87).

Pantalla submenú tipos de activos informáticos perfil activos fijos

Activos fijos SERTECPET

INGRESO DE TIPOS DE ACTIVOS

Descripcion:

Codigo	Descripcion	Cantidad
1	Laptop	
2	Monitor	
6	Impresora	
7	Access Point	
10	Workstation	
11	Servidor	
12	Access Point	
13	CPU	
14	Switch	
15	Firewall	

Opciones:

- ☐
- ☐
- ☐
- ☐

Figura 87. Pantalla submenú tipos de activos informáticos perfil activos fijos
Elaborado por: J. Navas y C. Álvarez

iv) Diagrama de procesos del perfil activos fijos (ver Figura 88).

Diagrama de Procesos perfil activos fijos

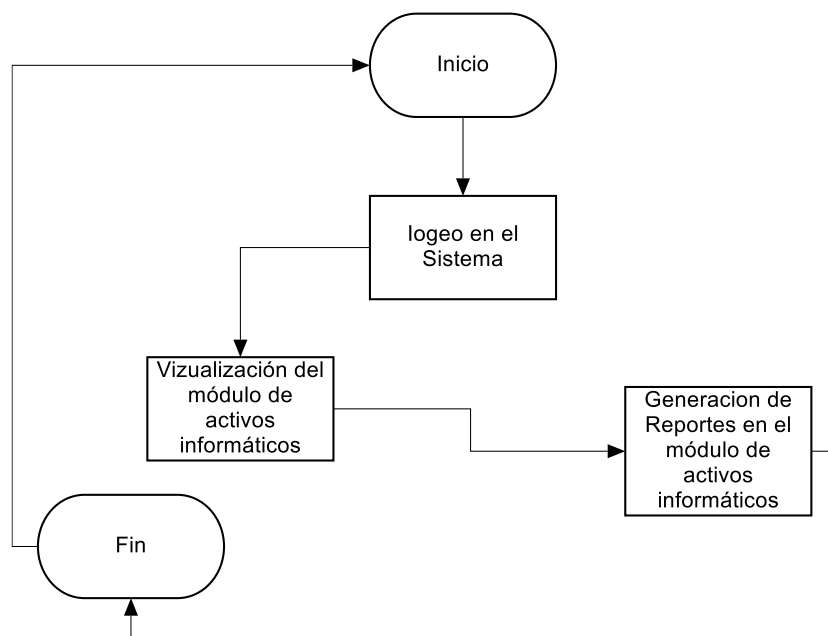


Figura 88. Diagrama de procesos perfil activos fijos
Elaborado por: J. Navas y C. Álvarez

4.5.5. Perfil de seguridad

- i) Pantalla principal del perfil de seguridad (ver Figura 89).

Pantalla principal perfil de seguridad



Figura 89. Pantalla principal perfil de seguridad
Elaborado por: J. Navas y C. Álvarez

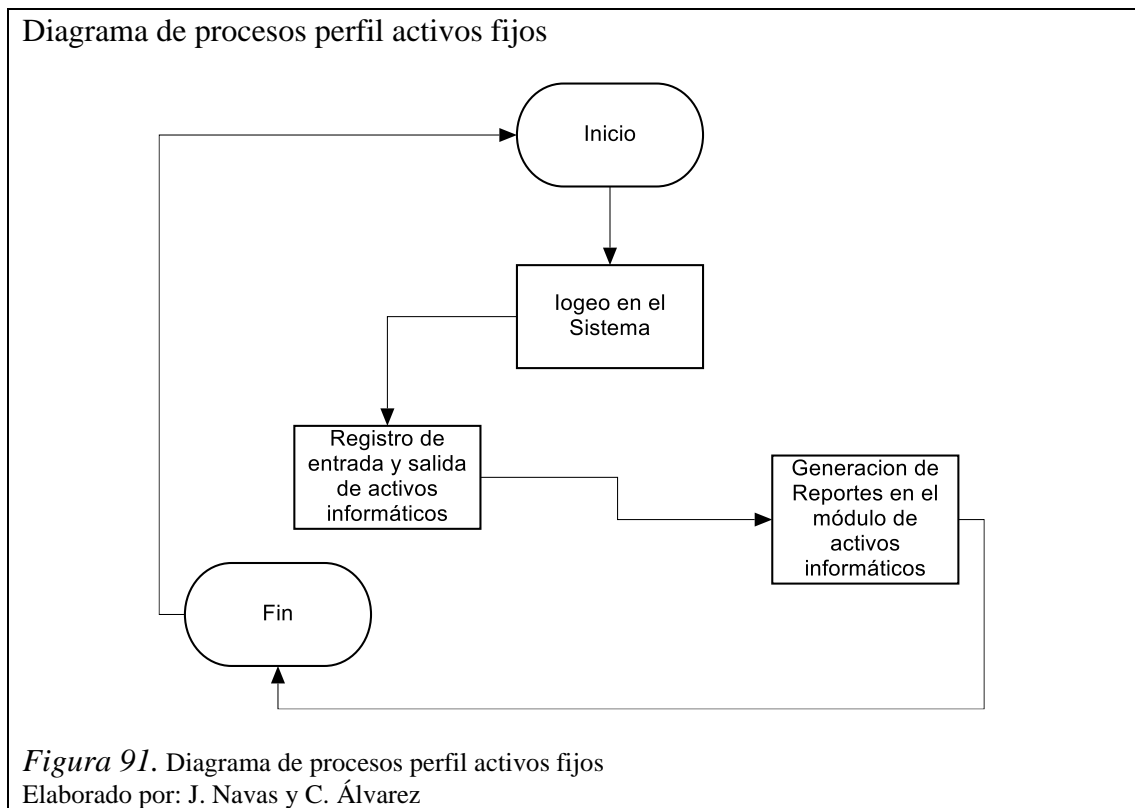
- ii) Pantalla de ingreso y salida perfil de seguridad (ver Figura 90).

Pantalla principal perfil de seguridad

Id	Fecha hora	Activo fijo	Movimiento	Cedula	Nombre

Figura 90. Pantalla principal perfil de seguridad
Elaborado por: J. Navas y C. Álvarez

iii) Diagrama de procesos del perfil de seguridad (ver Figura 91).



4.6. Instalación y ubicación de equipos

Dentro de la empresa los dispositivos de lectura RFID junto con el sistema fueron colocados en las máquinas de los guardias de seguridad y el administrador del sistema, estos fueron distribuidos de la siguiente manera:

1. Garita Eloy Alfaro en la cual se puede apreciar el lector RFID (dispositivo Color Negro) su cable de alimentación y cable de conexión con el equipo de la garita (ver Figura 92).

Guardia garita Eloy Alfaro



Figura 92. Guardia garita Eloy Alfaro
Elaborado por: J. Navas y C. Álvarez

2. Garita Arroyo de Rio en la cual se puede apreciar el lector RFID (dispositivo Color Negro) su cable de alimentación y cable de conexión con el equipo de la garita (ver Figura 93).

Guardia garita Arroyo del Rio



Figura 93. Guardia garita Arroyo del Rio
Elaborado por: J. Navas y C. Álvarez

3. Garita Campamento Base en la cual se puede apreciar el lector RFID (dispositivo Color Negro) su cable de alimentación y cable de conexión con el equipo de la garita (ver Figura 94).

Guardia garita campamento base



Figura 94. Guardia garita campamento base
Elaborado por: J. Navas y C. Álvarez

4. Departamento de sistemas administrador del sistema en la cual se puede apreciar el lector RFID (dispositivo Color Negro) su cable de alimentación y cable de conexión con el equipo del administrador (ver Figura 95).

Administrador del sistema



Figura 95. Administrador del sistema
Elaborado por: J. Navas y C. Álvarez

4.7. Análisis y resultados

Se han realizado guías de entrevista que se pueden observar en los anexos 5,6,7 posteriores a la implementación y uso del sistema a las personas involucradas inicialmente en este proyecto, en las cuales se puede apreciar que existe una mejoría en el control de los activos informáticos produciendo una aceptación por parte del coordinador de Activos Fijos el cual pudo evidenciar el proceso por el que se controla la entrada y salida de equipos en la garita (El guardia revisa al empleado, si posee equipo el mismo es registrado en el sistema, con hora y fecha).

Con la introducción del sistema el personal de seguridad posee ya una política por parte de la empresa para realizar un control más riguroso a los empleados.

El departamento de sistemas posee un mejor control en el movimiento y asignación de equipos a los empleados de la empresa facilitando así la ubicación de los mismos de manera inmediata mediante reportes generados en el sistema.

Para un levantamiento previo de la información de activos electrónicos que existían en la empresa se contrató a una consultora externa que realizo la recopilación de los datos, con un costo de \$5000 tomando un tiempo aproximado de 15 días para realizar los reportes (ver anexo 8).

Los resultados de mejora en cuanto a generación de información mediante los reportes obtenidos en el sistema comparados con los resultados anteriormente obtenidos por la consultora, se los calculó mediante hora/hombre tomando en cuenta al administrador del sistema de la siguiente manera:

Sueldo Administrador: \$1200

Tiempo de ejecución de reporte en el sistema: 20min (horario laboral)

Consultora: \$5000

Tiempo de ejecución de reportes: 15 días = 7200min (horario laboral)

Calculando el porcentaje de ejecución que toma realizar el reporte en el sistema actualmente, se evidencia que es 0.27% de tiempo en comparación al tiempo que le tomó a la consultora realizarlo.

$$\begin{aligned} 7200 &\leftarrow \rightarrow 100 \\ 20 &\leftarrow X \\ (20 \times 100) / 7200 &= 0,27\% \end{aligned}$$

El costo de desarrollo del sistema se lo cálculo de la siguiente manera:

Sueldo: \$1200

Tiempo empleado en la semana: 16 horas

Calculo de Hora laboral: \$1200/160 horas= \$7.50 hora

Calculo de tiempo realizado: 7 meses x 64 horas= 448 horas

Calculo de Costo aproximado del programa: \$7.50 x 448 horas = \$3360

Tabla 3. Comparación entre costo de la consultora y el sistema

Costo en dólares	Consultora	Sistema
	\$5000	\$3360

Nota: se ha realizado una comparación que tuvo el costo del sistema frente a nuestro proyecto, adaptado de la investigación, por J. Navas & C. Álvarez

Comparando el tiempo de levantamiento de información y de generación de reportes por parte de la empresa consultora que fue de 15 días contrastando con el tiempo que el sistema tarda un aproximado de 10 minutos, evidenciamos que existe una reducción de tiempo muy considerable. La reducción del costo para la empresa es igualmente evidenciable, ya que el valor por la contratación para el levantamiento de información y generación de reportes por la empresa consultora fue de \$5000 mientras que el costo total de la implementación del sistema fue de \$3360.

Adicional a estas cifras que evidencian el beneficio que trae a la empresa la implementación del sistema, se tienen algunas características adicionales como son la posibilidad de tener información constantemente actualizada y el conocer

en todo momento el movimiento que realizan los activos de entrada y salida de la empresa.

CONCLUSIONES

Mediante la implementación del sistema mejoro el tiempo de ejecución de reportes de inventario, comparando el tiempo que demoró la empresa consultora versus el sistema implementado tarda un 0,27% del tiempo que tomaba antes de la implementación del sistema.

Hay un ahorro significativo después de la puesta en marcha del sistema, antes de la implementación el costo para efectuar una recopilación de información y generación de reportes de inventario de activos electrónicos era de \$5000, actualmente el costo total del sistema es de \$3360.

Con la implementación del sistema se evita realizar nuevas contrataciones para la generación de reportes de activos informáticos ya que el sistema permite realizar reportes cuando se los requiera.

El área de seguridad física con la utilización del sistema tiene actualmente un procedimiento a seguir para la entrada y salida de los activos informáticos, teniendo así mayor control por parte del personal de seguridad en un periodo de 15 días de implementado el sistema no se han registrado pérdidas de los bienes de la compañía.

RECOMENDACIONES

Como proyecto contemplado se realizó una aplicación de modelo cliente servidor, para el futuro se podría aplicar una solución Web, la que permitiría tener acceso desde cualquier dispositivo mediante un browser accediendo así a las características de consulta que posee la aplicación mediante los perfiles.

También se podrían aplicar distintos tipos de etiquetas y lectores que se acoplarían al sistema, ya sea por su costo o factibilidad en el mercado actual, como por ejemplo código de barras, código QR, etc.

El proyecto en un futuro podría ser utilizado para otro tipo de activos, para ser utilizados en las bodegas de materiales o productos, además de otras locaciones que posea la empresa.

LISTA DE REFERENCIAS

- Arias, M. (16 de julio de 2012). *Java con Base de Datos*. Recuperado el 15 de septiembre de 2014, de javaconbasesdedatos.blogspot.com/2012_07_01_archive.html
- Dargan, G. (2004). *The Use of Radio Frequency Identification as a Replacement for Traditional Barcoding*. Chris.
- García de Jalón, J. (2000). *Aprenda Java como si estuviera en Primero*. San Sebastian, España: Tecnun.
- Microsoft. (2014). *Microsoft SQL Server 2008*. Recuperado el 15 de noviembre de 2014, de <http://www.microsoft.com/en-us/server-cloud/products/sql-server/#fbid=NZkDnWfADot>
- NX-ID. (2011). *Tecnología RFID*. Recuperado el 23 de septiembre de 2014, de <http://www.nx-id.com/web/tecnologiarrfid.php?lang=es&content=rfid>
- Silberschatz, A. (2002). *Fundamentos de Bases de Datos* (Cuarta ed.). Madrid, España: Mc GrawHill.
- Soluciones Pandaid. (8 de noviembre de 2012). *Etiquetas de RFID*. Recuperado el 18 de septiembre de 2014, de <http://www.pandaid.com/que-es-una-etiqueta-rfid/>
- Solutions Tadic. (2014). *Tecnología RFID*. Recuperado el 19 de septiembre de 2014, de <http://www.tadic-solutions.com/rfid/>
- Somerville, I. (2005). *Ingeniería del Software* (Séptima ed.). Madrid, España: Pearson.
- Tanenbaum, A. S. (2011). *Computers Networks* (Quinta ed.). Boston, Estados Unidos: Pearson.
- Texas Instruments. (2012). *Tecnología RFID*. Recuperado el 24 de octubre de 2014, de <http://www.ti.com/rfid/docs/news/eNews/enewsvol43b.shtml>
- Universidad Técnica de Machala. (4 de septiembre de 2013). *Presentación de proyecto de programación*. Recuperado el 22 de noviembre de 2014, de <http://www.slideshare.net/jeffer379/presentacion-25873333>

Vásquez, A. (2012). *Manual de SQL-Server-2008*. Recuperado el 15 de noviembre de 2014, de <http://www.youblisher.com/p/190598-Manual-SQL-Server-2008/>

ANEXOS

Anexo 1. Guía de entrevista realizada al coordinador de activos fijos

¿Existe alguna política sobre los activos fijos en la empresa?

En la empresa se considera activos fijos todo aquello mayor o igual a \$700, que representan un bien directo para la compañía, adicional a esto existe una subcategoría denominada activos de control todo aquello comprendido entre \$100 y \$300 lo que no representa un gasto significativo para la compañía, todo aquello menor a \$100 es considerado suministro.

¿Cómo lleva el control de activos el área de activos?

Cuando un bien se ingresa a la compañía debe pasar por obligación por el departamento de activos para su ingreso al sistema que posee la empresa (Microsoft AX 2009), posteriormente si es un equipo electrónico se traslada al departamento de TIC o al departamento que corresponda.

¿Qué sucede cuando un equipo no se encuentra en la empresa, cual es el método que se aplica?

Lamentablemente no existe un control sobre la salida de equipos a excepción de la guardianía que realiza una revisión en los accesos de entrada y salida, toma el número de activo del dispositivo y lo anotan en un papel, esta información no se refleja en ningún lugar adicional ni se almacena.

¿Si existiera la pérdida de un activo, que se realiza?

Se procede a realizar una auditoría del bien al departamento asignado dicho activo. Cada departamento debe tener a cargo una persona que genere actas de entrega que permitan tener responsables de los activos dentro de la empresa, y remitir a un responsable para generar el descuento al Rol si es el caso.

¿En su opinión cual es la necesidad actual que se tiene en su departamento para un control eficiente de los activos?

Como departamento tenemos la necesidad de conocer la información de los activos como por ejemplo: Su localización, a qué persona fue asignada, status del Activo, descripción del activo, etc. Por lo que sería óptimo poseer un sistema informático en el cual permita emitir reportes sobre la información anteriormente descrita.

Anexo 2. Guía de entrevista realizada al coordinador de TI

¿Cuál es el proceso que se realiza cuando ingresa un activo informático al departamento de TI?

Si un activo informático entra al departamento de TI, en primer lugar hay que prepararlo, es decir instalando los programas base que se utiliza dentro de la empresa, si se requiere algún otro programa para su instalación se lo realiza mediante el pedido del departamento al que se le va a asignar dicho equipo, para realizar la descarga del equipo fuera del departamento de TI se necesita generar un acta de entrega esta se la realiza de manera manual alimentando los datos en un pequeño modulo del sistema Microsoft Ax 2009, se imprime el reporte se procede a las firmas tanto de la persona que recibe como la que entrega así como la firma de aprobación del Jefe de TI.

¿Cree usted que se lleva un control efectivo de los activos informáticos?

Lamentablemente no se puede llevar un control exhaustivo del equipo asignado, nosotros nos remitimos a las actas en caso de alguna perdida, daño, o mal uso del mismo, en muchas ocasiones, los equipos salen de la empresa sin autorización del Jefe de cada departamento.

En las oficinas que poseemos en las provincia de Orellana en la ciudad del Coca tenemos un mayor inconveniente debido a que existe una gran afluencia de salida y entrada de equipos hacia los diferentes pozos, no hay un control seguro y efectivo que nos dé la certeza de que los equipos que salen van a regresar, la guardianía realiza una revisión sencilla al personal en ciertas ocasiones, lo cual no asegura que los mismos lleven consigo un equipo, esta información no es transmitida ni al departamento de Activos Fijos ni al de TI.

¿Cuál sería una alternativa para tener un mejor control de los activos informáticos?

Sería ideal poseer un sistema de control de activos informáticos, el cual permita tener una información actualizada de los equipos, es decir conocer el momento exacto en el que un dispositivo abandona las instalaciones de la empresa, a su vez con la

información almacenada y disponible se puede generar actas de entrega, inventarios, reasignaciones, etc.

¿Ha escuchado de la tecnología RFID?

Si me parece interesante este tipo de tecnología que se ha utilizado en grandes empresas, facilitando el etiquetamiento de activos a nivel mundial como es el caso de DHL, TC televisión entre otras, sería de gran utilidad si esta tecnología es aplicada al sistema que se requiere en la empresa.

Anexo 3. Guía de entrevista realizada al supervisor de seguridad

¿Existe alguna política sobre el uso de equipos fuera de la empresa?

Como seguridad nosotros nos encargamos de que los equipos que abandonen la empresa sean registrados mediante un “cacheo” al personal que salga de la misma.

¿Cómo lleva el control de equipos el personal de seguridad?

Actualmente los equipos registrados son anotados en un cuaderno que lleva el historial de asistencias del personal que labora en el empresa, en el cual es registrado hora de entrada y salida, si el empleado procede a sacar un equipo de las oficinas se lo registra en base a su nombre adicionando N° de Activo y Marca del equipo.

¿Qué sucede con esta información?

La parte de entrada y salida es entregada al personal de Talento Humano que se encargan de los descuentos y amonestaciones si fuera el caso, por otra parte los equipos no tienen ningún registro por parte de la empresa por lo que lo registrado no tiene “mucho peso”.

¿Si existiera la pérdida de un activo, que se realiza?

Se procedería a realizar una búsqueda en el historial de anotaciones según el N° de Activo y Marca del equipo, lamentablemente no es un medio confiable por lo que los equipos no presentan un historial a la mano que puede facilitar el trabajo.

¿En su opinión cual es la necesidad actual que se tiene para un control eficiente de los equipos?

Como seguridad tenemos la necesidad de registrar los equipos y que estos sean validados por personal de la empresa, que se lleve un seguimiento de los mismos para mantenernos informados y darnos una mano para evitar inconvenientes

Anexo 4. Levantamiento de Información

Nombre	Cedula	Equipo	Modelo	Marca	Memoria	Procesador	Serie	N° Activo	Nombre del Equipo	Sistema Operativo
INGENIERIA										
Alex García	180374993 4	PC	DH55PJ	CLON	8192MB	Intel(R) Core(TM) i7 CPU 870 @ 2.93GHz (8 CPUs), ~2.9GHz	S/N	S/N	ECUIOWD01383	Windows 7 Professional 64-bit (6.1, Build 7601) Service Pack 1
Álvaro Penaherrera	171370616 4	PC	DH55PJ	CLON	8192MB	Intel(R) Core(TM) i7 CPU 870 @ 2.93GHz (8 CPUs), ~2.9GHz	S/N	2021112	ECUIOWD01401	Windows 7 Professional 64-bit (6.1, Build 7601) Service Pack 1
Alejandro Salcedo	171583061 6	Laptop	EliteBook 8570w	HP	8192MB	Intel(R) Core(TM) i7- 3630QM CPU @ 2.40GHz (8 CPUs), ~2.4GHz	5CB2482FKL	2038335	ECUIOWL01894	Windows 7 Professional 64-bit (6.1, Build 7601) Service Pack 1

Brumel Suarez	1707330153	Laptop	EliteBook 8460p	HP	8192MB	Windows 7 Professional 64-bit (6.1, Build 7601) Service Pack 1	107LTVB0G935	2036140	ECUIOWL01566	Windows 7 Professional 64-bit (6.1, Build 7601) Service Pack 1
Diana Bustos	502227168	PC	Compaq 6000 Pro SFF PC	HP	4096MB	Intel(R) Core(TM)2 Quad CPU Q8400 @ 2.66GHz (4 CPUs), ~2.7GHz	MXJ95303ZS	2036232	ECUIOWD01753	Windows 7 Professional 64-bit (6.1, Build 7601) Service Pack 1
Deán Aguas	1204320095	PC	Compaq dc5700 Microtower	HP	2048MB	Intel(R) Core(TM)2 CPU 6400 @ 2.13GHz (2 CPUs), ~2.1GHz	S/N	S/N	ECUIOWD01746	Windows 7 Professional 64-bit (6.1, Build 7601) Service Pack 1
David Garzón	1710634674	Laptop	EliteBook 8470p	HP	4096MB	Intel(R) Core(TM) i7-3520M CPU @ 2.90GHz (4 CPUs), ~2.9GHz	CNU329C195	2036393	ECUIOWL001690	Windows 7 Professional 64-bit (6.1, Build 7601) Service Pack 1

David Montaluisa	502658461	Laptop	DH55PJ	CLON	8192MB	Intel(R) Core(TM) i7 CPU 870 @ 2.93GHz (8 CPUs), ~2.9GHz	S/N	S/N	ECUIOWD00327	Windows 8 Pro 64-bit (6.2, Build 9200)
Eduardo González	1708058779	Workstation	T7400	Dell	16384MB	Intel(R) Xeon(R) CPU X5472 @ 3.00GHz (4 CPUs), ~3.0GHz	CNU329C197	2036317	ECUIOWL00365	Windows 8 Pro 64-bit (6.2, Build 9200)
Héctor Arévalo	1713217162	Laptop	ProBook 4440s	HP	8192MB	Intel(R) Core(TM) i5-3210M CPU @ 2.50GHz (4 CPUs), ~2.5GHz	2CE2361BC6	1858	ECUIOWL01858	Windows 7 Professional 64-bit (6.1, Build 7601)
Héctor Rangel	1754493961	Laptop	EliteBook 8440p	HP	4096MB	Intel(R) Core(TM) i7 CPU M 640 @ 2.80GHz (4 CPUs), ~2.8GHz	CND0470WLD	2037628	ECUIOWL01373	Windows 7 Professional 64-bit (6.1, Build 7601)
Hsys	1713217162	Laptop	M17xR3	Alienware	16384MB	Intel(R) Core(TM) i7-2630QM CPU @ 2.00GHz (8 CPUs), ~2.0GHz	6XVN5Q1	1475	ECUIOWL01475	Windows 7 Professional 64-bit (6.1, Build 7600)

Jorge Aldas	180193859 6	Laptop	Precision M4800	Dell	16384MB	Intel(R) Core(TM) i7- 4600M CPU @ 2.90GHz (4 CPUs), ~2.9GHz	1RRJXZ1	2080198	ECUIOWL20801 98	Windows 7 Professional 64-bit (6.1, Build 7601) Service Pack 1
Jairo Chávez	171593806 2	PC	Compaq Elite 8300	HP	8192MB	Intel(R) Core(TM) i7- 3770 CPU @ 3.40GHz (8 CPUs), ~3.4GHz	MXL 2340J4J	2040420	ECUIOWD0349	Windows 7 Professional 64-bit (6.1, Build 7601)
Juan Carlos Vallejo	603609371	Laptop	EliteBook 8470p	HP	4096MB	Intel(R) Core(TM) i7- 3520M CPU @ 2.90GHz (4 CPUs), ~2.9GHz	CNU3019FBL	1943	ECUIOWL20695 68	Windows 7 Professional 64-bit (6.1, Build 7601)
Juan Carlos Vallejo	603609371	PC	DH55PJ	CLON	8192MB	Intel(R) Core(TM) i7 CPU 870 @ 2.93GHz (8 CPUs), ~2.9GHz	S/N	1389	ECUIOWD00389	Windows 8 Pro 64-bit (6.2, Build 9200)
Javier Guzmán	171472258 2	Laptop	ProBook 4430s	HP	4096MB	Intel(R) Core(TM) i5- 2410M CPU @ 2.30GHz (4 CPUs), ~2.3GHz	Borrado	2037109	ECUIOWL01518	Windows 7 Professional 64-bit (6.1, Build 7601)

Jenny Simbaña	171089138 1	Laptop	EliteBook 8470p	HP	4096MB	Intel(R) Core(TM) i7- 3520M CPU @ 2.90GHz (4 CPUs), ~2.9GHz	CNU329C177	2036201	ECUIOWL00169 1	Windows 7 Professional 64-bit (6.1, Build 7601)
Juan Villavicencio	160036075 2	Laptop	EliteBook 8470p	HP	4096MB	Intel(R) Core(TM) i7- 3520M CPU @ 2.90GHz (4 CPUs), ~2.9GHz	MXL3021SRX	2036355	ECUIOWL20693 60	Windows 7 Professional 64-bit (6.1, Build 7601)
Miguel Hidalgo	171938467 7	Laptop	Portege Z930	Toshiba	6144MB	Intel(R) Core(TM) i5- 3317U CPU @ 1.70GHz (4 CPUs), ~1.7GHz	4D0566559H	2039592	ECOCCWL00168 2	Windows 7 Professional 64-bit (6.1, Build 7601)
Neydi Niacasha	802146423	PC	Compaq Elite 8300 CMT	HP	4096MB	Intel(R) Core(TM) i7- 3770 CPU @ 3.40GHz (8 CPUs), ~3.4GHz	MXL2490G92	2037222	ECUIOWD01916	Windows 7 Professional 64-bit (6.1, Build 7601)
Pasante	_____	PC	Compaq Elite 8300 CMT	HP	4096MB	Intel(R) Core(TM) i5- 3470 CPU @ 3.20GHz (4 CPUs), ~3.2GHz	MXL4670L94	2034467	ECUIOWL20363 55	Windows 7 Professional 64-bit (6.1, Build 7601)

Sebastian Arroyo	1714661855	Laptop	ProBook 4420s	HP	3072MB	Intel(R) Core(TM) i3 CPU M 370 @ 2.40GHz (4 CPUs), ~2.4GHz	CNF0508W9W	2037246	ECUIOWL01460	Windows 7 Professional 32-bit (6.1, Build 7601)
Stalin Tipan	1714681887	Laptop	EliteBook 8470p	HP	4096MB	Intel(R) Core(TM) i7-3520M CPU @ 2.90GHz (4 CPUs), ~2.9GHz	CNU3019F56	2037390	ECUIOWL01925	Windows 7 Professional 64-bit (6.1, Build 7601)
Thomas Martínez	1600195448	Laptop	EliteBook 8470p	HP	4096MB	Intel(R) Core(TM) i7-3520M CPU @ 2.90GHz (4 CPUs), ~2.9GHz	CNU329C17L	2071035	ECUIOWL01723	Windows 7 Professional 64-bit (6.1, Build 7601)
Wilson Vásquez	1718242983	Laptop	EliteBook 8440p	HP	4096MB	Intel(R) Core(TM) i7 CPU M 640 @ 2.80GHz (4 CPUs), ~2.8GHz	CND047R7B	2037680	ECUIOWL01375	Windows 7 Professional 64-bit (6.1, Build 7601)
Ximena Martínez	1720488251	Laptop	Pavilion 14 Notebook PC	HP	16384MB	Intel(R) Core(TM) i7-4702MQ CPU @ 2.20GHz (8 CPUs), ~2.2GHz	CNU329C1B1	2036270	ECUIOWL01718	Windows 8 Pro 64-bit (6.2, Build 9200)
COMPRAS										

Aracely Cruz	1718829359	Laptop	Latitude E6440	Dell	4096MB	Intel(R) Core(TM) i7-4600M CPU @ 2.90GHz (4 CPUs), ~2.9GHz	JRYGVZ1	2047177	ECUIOWL001730	Windows 7 Professional 64-bit (6.1, Build 7601)
Cesar Madrid	801856964	Laptop	ProBook 4440s	HP	4096MB	Intel(R) Core(TM) i5-3210M CPU @ 2.50GHz (4 CPUs), ~2.5GHz	2CE22726C1	2047276	ECUIOWL2047276	Windows 7 Professional 64-bit (6.1, Build 7601)
Diana Constante	1721982666	Laptop	PORTEGE Z930	Toshiba	6144MB	Intel(R) Core(TM) i5-3317U CPU @ 1.70GHz (4 CPUs), ~1.7GHz	2DOA34512	2047436	ECUIOWL2047436	Windows 8 Pro 64-bit (6.2, Build 9200)
Giovanny Maldonado	1713037941	Laptop	Latitude E6440	Dell	4096MB	Intel(R) Core(TM) i7-4600M CPU @ 2.90GHz (4 CPUs), ~2.9GHz	2M7RWZ1	2080426	ECUIOWL2080426	Windows 7 Professional 64-bit (6.1, Build 7601)
Henry Silva	201570165	Laptop	Compaq 6530b	HP	2048MB	Intel(R) Core(TM)2 Duo CPU P8700 @ 2.53GHz (2 CPUs), ~2.5GHz	CNU0110ZM9	1238	ECUIOWL01238	Windows 7 Professional 64-bit (6.1, Build 7601)

José Sevilla	1706862628	Laptop	Latitude E6440	Dell	4096MB	Intel(R) Core(TM) i7-4600M CPU @ 2.90GHz (4 CPUs), ~2.9GHz	D1KRVZ1	2069377	ECUIOWL2069377	Windows 7 Professional 64-bit (6.1, Build 7601)
Lucia Zapata	1710867142	Laptop	ProBook 4440s	HP	4096MB	Intel(R) Core(TM) i5-3210M CPU @ 2.50GHz (4 CPUs), ~2.5GHz	2CE24508S1	2047719	ECUIOWL01890	Windows 7 Professional 64-bit (6.1, Build 7601)
María Fernanda Cruz	1714042338	Laptop	PORTEGE Z930	Toshiba	6144MB	Intel(R) Core(TM) i5-3317U CPU @ 1.70GHz (4 CPUs), ~1.7GHz	ZC018849H	2047580	ECUIOWL01915	Windows 7 Professional 64-bit (6.1, Build 7601)
Mario Jurado	1713539615	Laptop	Latitude E6440	Dell	4096MB	Intel(R) Core(TM) i7-4600M CPU @ 2.90GHz (4 CPUs), ~2.9GHz	CNF04476CW	2080242	ECUIOWL2080242	Windows 7 Professional 64-bit (6.1, Build 7601)
María Penaherrera	1713767687	Laptop	ProBook 4540s	HP	4096MB	Intel(R) Core(TM) i5-3210M CPU @ 2.50GHz (4 CPUs), ~2.5GHz	2CE24020RY	2047399	ECUIOWL2047399	Windows 7 Professional 64-bit (6.1, Build 7601)

Sheila Chávez	171118758 1	Laptop	ProBook 4540s	HP	4096MB	Intel(R) Core(TM) i5- 3210M CPU @ 2.50GHz (4 CPUs), ~2.5GHz	2CE24100KK	1877	ECUIOWL01817	Windows 7 Professional 64-bit (6.1, Build 7601)
William Villafuerte	171761302 8	PC	Compaq Pro 6300 MT	HP	4096MB	Intel(R) Core(TM) i5- 3470 CPU @ 3.20GHz (4 CPUs), ~3.2GHz	MXL24202FV	2047542	ECUIOWD01806	Windows 7 Professional 64-bit (6.1, Build 7601)
FINANZAS										
Alejandra Mosquera	171656201 0	PC	Compaq Pro 6300 SFF	HP	4096MB	Intel(R) Core(TM) i5- 3470 CPU @ 3.20GHz (4 CPUs), ~3.2GHz	MXL23613RB	2044473	ECUIOWD1815	Windows 7 Professional 64-bit (6.1, Build 7601)
Andrea Soria	171900260 0	PC	Compaq Pro 6300 MT	HP	4096MB	Intel(R) Core(TM) i5- 3470 CPU @ 3.20GHz (4 CPUs), ~3.2GHz	MXL2412NF	2044763	ECUIOWL00026 9	Windows 7 Professional 64-bit (6.1, Build 7601)
Adriana Zambonino	603600909	Laptop	ProBook 4540s	HP	4096MB	Intel(R) Core(TM) i5- 3210M CPU @ 2.50GHz (4 CPUs), ~2.5GHz	2CE24100JB	2044701	ECUIOWL01882	Windows 7 Professional 64-bit (6.1, Build 7601)

Claudia Baquero	171435553 2	CPU	DH55PJ	CLON	8192MB	Intel(R) Core(TM) i7 CPU 870 @ 2.93GHz (8 CPUs), ~2.9GHz	S/N	2044428	ECUIOWD00294	Windows 8 Pro 64-bit (6.2, Build 9200)
Cesar Gordillo	800407124	CPU	Compaq Elite 8300 CMT	HP	4096MB	Intel(R) Core(TM) i7- 3770 CPU @ 3.40GHz (8 CPUs), ~3.4GHz	MXL2490J1D	2037857	ECUIOWD01897	Windows 7 Professional 64-bit (6.1, Build 7601)
Claudia Corral	501804678	CPU	Compaq Elite 8300 CMT	HP	4096MB	Intel(R) Core(TM) i7- 3770 CPU @ 3.40GHz (8 CPUs), ~3.4GHz	MXL2490G8G	299	ECUIOWD01898	Windows 7 Professional 64-bit (6.1, Build 7601)
Damián Acosta	171591983 1	CPU	Compaq DC5700 Microtower	HP	2048MB	Intel(R) Core(TM)2 CPU 6400 @ 2.13GHz (2 CPUs), ~2.1GHz	MXJ73808GB	2038120	ECUIOWD01736	Windows 7 Professional 32-bit (6.1, Build 7601)
Daniel Torres	171188353 6	Laptop	ProBook 4420s	HP	3072MB	Intel(R) Core(TM) i3 CPU M 370 @ 2.40GHz (4 CPUs), ~2.4GHz	CNF0508W97	1408	ECUIOWL01408	Windows 7 Professional 32-bit (6.1, Build 7601)

Daniel Vallejo	171678614 8	Laptop	EliteBook 8440p	HP	4096MB	Intel(R) Core(TM) i7 CPU M 640 @ 2.80GHz (4 CPUs), ~2.8GHz	S/N	1484	ECUIOWL01494	Windows 7 Professional 32-bit (6.1, Build 7600)
Erika Almeida	171966925 9	PC	Compaq 6000 Pro MT PC	HP	2048MB	Intel(R) Core(TM)2 Quad CPU Q8400 @ 2.66GHz (4 CPUs), ~2.7GHz	MXL1070K2N	253	ECUIOWD01853	Windows 7 Professional 32-bit (6.1, Build 7601)
Henry Uyaguari	171759463 2	Laptop	XPS M1330	Dell	2048MB	Intel(R) Core(TM)2 Duo CPU T7250 @ 2.00GHz (2 CPUs), ~2.0GHz	8Q8FQG1	2044640	ECUIOWL20446 40	Windows 7 Professional 64-bit (6.1, Build 7601)
Karen Córdova	220007673 1	Laptop	ProBook 4420s	HP	2048MB	Intel(R) Core(TM) i3 CPU M 370 @ 2.40GHz (4 CPUs), ~2.4GHz	CNF04476CW	2039516	ECUIOWL01447	Windows 7 Professional 32-bit (6.1, Build 7601)
María Elena Morejón	170797034 7	PC	OptiPlex 7010	Dell	4096MB	Intel(R) Core(TM) i5- 3470 CPU @ 3.20GHz (4 CPUs), ~3.2GHz	4TBMRV1	1653	ECUIOWD00165 3	Windows 7 Professional 64-bit (6.1, Build 7601)

Maritza Esparza	171191066 9	Laptop	ProBook 4430s	HP	4096MB	Intel(R) Core(TM) i5- 2450M CPU @ 2.50GHz (4 CPUs), ~2.5GHz	CNU207269C	2037901	ECUIOWL00011 3	Windows 7 Professional 32-bit (6.1, Build 7601)
Marcelo García	172308030 3	PC	Compaq dc5700 Microtower	HP	2048MB	Intel(R) Core(TM)2 CPU 6400 @ 2.13GHz (2 CPUs), ~2.1GHz	MXJ8020C5Z	2044596	ECUIOWD01732	Windows 7 Professional 64-bit (6.1, Build 7601)
Milton Vasconez	171645408 5	Laptop	PORTEGE Z930	Toshiba	6144MB	Intel(R) Core(TM) i5- 3317U CPU @ 1.70GHz (4 CPUs), ~1.7GHz	2C0189834	2036737	ECUIOWL01908	Windows 7 Professional 64-bit (6.1, Build 7601)
María Muñoz	172024276 5	PC	Compaq 6000 Pro MT PC	HP	4096MB	Intel(R) Core(TM)2 Quad CPU Q8400 @ 2.66GHz (4 CPUs), ~2.7GHz	MXL0250GC9	2044367	ECUIOWD02073	Windows 7 Professional 64-bit (6.1, Build 7601)
María Orellana	172109489 2	Laptop	Latitude E6440	Dell	4096MB	Intel(R) Core(TM) i7- 4600M CPU @ 2.90GHz (4 CPUs), ~2.9GHz	GPVYXZ1	2080259	ECUIOWL20802 59	Windows 7 Professional 64-bit (6.1, Build 7601)

GESTION DE PROCESOS										
Darío Ramírez	180343835 5	Laptop	Latitude E6440	Dell	4096MB	Intel(R) Core(TM) i7- 4600M CPU @ 2.90GHz (4 CPUs), ~2.9GHz	B2ZGVZ1	1731	ECUIOWL01731	Windows 7 Professional 64-bit (6.1, Build 7601)
Edwin Hidalgo	171146697 7	Laptop	ProBook 4420s	HP	4096MB	Intel(R) Core(TM) i3 CPU M 370 @ 2.40GHz (4 CPUs), ~2.4GHz	CNF048H0P	1856	ECUIOWL01856	Windows 7 Professional 32-bit (6.1, Build 7601)
Evelyn Lucero	171152278 7	Laptop	ProBook 4430s	HP	4096MB	Intel(R) Core(TM) i3- 2310M CPU @ 2.10GHz (4 CPUs), ~2.1GHz	S/N	1477	ECUIOWL01477	Windows 7 Professional 64-bit (6.1, Build 7601)
Franklin Chiluisa	171179448 5	Laptop	EliteBook 8470p	HP	4096MB	Intel(R) Core(TM) i7- 3540M CPU @ 3.00GHz (4 CPUs), ~3.0GHz	CNU406956W	2069285	ECUIOWL20692 85	Windows 7 Professional 64-bit (6.1, Build 7601)
OPERACIONES										
Ángel Ushiña	170525290 4	Laptop	Pavilion 14 Notebook PC	HP	8192MB	Intel(R) Core(TM) i7- 4702MQ CPU	5CD543412367	2071165	ECUIOWL01714	Windows 8 Pro 64-bit (6.2, Build 9200)

						@ 2.20GHz (8 CPUs), ~2.2GHz				
Betty Ledesma	1500434947	Laptop	Latitude E6440	Dell	4096MB	Intel(R) Core(TM) i7-4600M CPU @ 2.90GHz (4 CPUs), ~2.9GHz	8TXRWZ1	2080075	ECUIOWL2080075	Windows 7 Professional 64-bit (6.1, Build 7601)
Erik Miranda	1710626902	Laptop	ProBook 4420s	HP	3072MB	Intel(R) Core(TM) i5 CPU M 460 @ 2.53GHz (4 CPUs), ~2.5GHz	Borrada	585	ECUIOWL01329	Windows 7 Professional 32-bit (6.1, Build 7601)
Jefferson Ninabanda	1716594930	Laptop	Latitude E6440	Dell	4096MB	Intel(R) Core(TM) i7-4600M CPU @ 2.90GHz (4 CPUs), ~2.9GHz	GQLRWZ1	2080129	ECUIOWL2080129	Windows 7 Professional 64-bit (6.1, Build 7601)
Julissa Hurtado	1716976046	Laptop	Compaq 6730b	HP	2048MB	Intel(R) Core(TM)2 Duo CPU P8600 @ 2.40GHz (2 CPUs), ~2.4GHz	I6242L3456	2069346	ECUIOWL2069346	Windows 7 Professional 64-bit (6.1, Build 7601)
Juan Zambrano	1715834022	Laptop	nc75xx	HP	2048MB	Intel(R) Core(TM)2 Duo CPU T7500 @ 2.20GHz (2 CPUs), ~2.2GHz	CNU7341Q6M	2074418	ECUIOWL0650	Windows 7 Professional 64-bit (6.1, Build 7601)

Luis Llerena	170871799 4	Laptop	Pavilion 14 Notebook PC	HP	8192MB	Intel(R) Core(TM) i7- 4702MQ CPU @ 2.20GHz (8 CPUs), ~2.2GHz	5CD3342P7W	2028640	ECUIOWL01708	Windows 8 Pro 64-bit (6.2, Build 9200)
Mario Muso	180080094 6	Laptop	Latitude E6420	Dell	4096MB	Intel(R) Core(TM) i5- 2540M CPU @ 2.60GHz (4 CPUs), ~2.6GHz	Borrado	1601	ECUIOWL01501	Windows 7 Professional 64-bit (6.1, Build 7601)
William Medina	171523150 0	Laptop	Latitude E6440	Dell	4096MB	Intel(R) Core(TM) i7- 4600M CPU @ 2.90GHz (4 CPUs), ~2.9GHz	BFJRVZ1	2069407	ECOCCWL20694 07	Windows 7 Professional 64-bit (6.1, Build 7601)
GESTION DE PROYECTOS										
Cristian Maldonado	171546588 4	Laptop	EliteBook 8470p	HP	4096MB	Intel(R) Core(TM) i7- 3520M CPU @ 2.90GHz (4 CPUs), ~2.9GHz	CNU237BCHZ	2069025	ECUIOWL01861	Windows 7 Professional 64-bit (6.1, Build 7601)
Gino Hinojosa	170856321 6	Laptop	ProBook 4440s	HP	4096MB	Intel(R) Core(TM) i5- 3210M CPU @ 2.50GHz (4 CPUs), ~2.5GHz	2CE236187P	2069094	ECUIOWL01865	Windows 7 Professional 64-bit (6.1, Build 7601)

Gina Loachamin	1713707105	Laptop	PORTEGE Z930	Toshiba	6144MB	Intel(R) Core(TM) i5-3317U CPU @ 1.70GHz (4 CPUs), ~1.7GHz	2D035049H	1618	ECUIOWL01618	Windows 7 Professional 64-bit (6.1, Build 7601)
Luis Álvarez	1718162181	Laptop	EliteBook 8470p	HP	4096MB	Intel(R) Core(TM) i7-3520M CPU @ 2.90GHz (4 CPUs), ~2.9GHz	CNU329C18S	2070786	ECUIOWL2070786	Windows 7 Professional 64-bit (6.1, Build 7601)
AUDITORES										
Jose Suarez	1724357734	Laptop	Latitude E6440	Dell	4096MB	Intel(R) Core(TM) i7-4600M CPU @ 2.90GHz (4 CPUs), ~2.9GHz	3.371E+09	2069599	ECUIOWL2069599	Windows 7 Professional 64-bit (6.1, Build 7601)
Luis Zarate	1716840184	Laptop	PORTEGE Z930	Toshiba	6144MB	Intel(R) Core(TM) i5-3317U CPU @ 1.70GHz (4 CPUs), ~1.7GHz	3D052150H	2068776	ECUIOWL001626	Windows 7 Professional 64-bit (6.1, Build 7601)
Orlando Jerez	1720086030	Laptop	PORTEGE Z930	Toshiba	6144MB	Intel(R) Core(TM) i5-3317U CPU @	3D052250H	2068251	ECUIOWL001630	Windows 7 Professional 64-bit (6.1, Build 7601)

						1.70GHz (4 CPUs), ~1.7GHz				
PROYECTOS										
Daniel Galarza	1720175890	Laptop	TECRA R940	Toshiba	12288MB	Intel(R) Core(TM) i7-3540M CPU @ 3.00GHz (4 CPUs), ~3.0GHz	YDO35212H	2069650	ECUIOWL2069650	Windows 7 Professional 64-bit (6.1, Build 7601)
Elizabeth Zambrano	1718126350	Laptop	Latitude E6440	Dell	4096MB	Intel(R) Core(TM) i7-4600M CPU @ 2.90GHz (4 CPUs), ~2.9GHz	96RRWZ1	2069476	ECUIOWL2069476	Windows 7 Professional 64-bit (6.1, Build 7601)
Gabriela Pillajo	1721484184	Laptop	TECRA R940	Toshiba	12288MB	Intel(R) Core(TM) i7-3540M CPU @ 3.00GHz (4 CPUs), ~3.0GHz	YD035214H	2069643	ECUIOWL2069643	Windows 7 Professional 64-bit (6.1, Build 7601)
Lucy Velepucha	701988214	Laptop	PORTEGE Z930	Toshiba	6144MB	Intel(R) Core(TM) i5-3317U CPU @ 1.70GHz (4 CPUs), ~1.7GHz	S/N	1683	ECUIOWL01683	Windows 7 Professional 64-bit (6.1, Build 7601)
Nataly Silva	1723566012	Laptop	Vostro 270s	Dell	6144MB	Intel(R) Core(TM) i5-	DL1BQW1	1929	ECUIOWD01929	Windows 8 Pro 64-bit

						3450S CPU @ 2.80GHz (4 CPUs), ~2.8GHz				(6.2, Build 9200)
Paola Hidrovo	1712567583	Laptop	PORTEGE Z930	Toshiba	6144MB	Intel(R) Core(TM) i5-3317U CPU @ 1.70GHz (4 CPUs), ~1.7GHz	4D051859H	2038326	ECUIOWL01694	Windows 7 Professional 64-bit (6.1, Build 7601)
GRAN TIERRA										
Christopher León	1714261557	CPU	ProDesk 600 G1 SFF	HP	16384MB	Intel(R) Core(TM) i7-4770 CPU @ 3.40GHz (8 CPUs), ~3.4GHz	83406065	2038106	ECUIOWD01709	Windows 8 Pro 64-bit (6.2, Build 9200)
Fredy Toapanta	502664493	Laptop	EliteBook 8770w	HP	8192MB	Intel(R) Core(TM) i7-3720QM CPU @ 2.60GHz (8 CPUs), ~2.6GHz	CNU236B68V	1901	ECUIOWL01901	Windows 7 Professional 64-bit (6.1, Build 7601)
Irina Viera	502846967	CPU	ProDesk 600 G1 SFF	HP	16384MB	Intel(R) Core(TM) i7-4770 CPU @ 3.40GHz (8 CPUs), ~3.4GHz	CNUB20GVF	1712	ECUIOWD01712	Windows 7 Professional 64-bit (6.1, Build 7601)
Karen Romero	1104335060	Laptop	EliteBook 8470p	HP	8192MB	Intel(R) Core(TM) i7-3520M CPU @	CNU329C18P	1692	ECUIOWL2040581	Windows 7 Professional 64-bit (6.1, Build 7601)

						2.90GHz (4 CPUs), ~2.9GHz				
Luis Villagómez	1713840963	Laptop	ProBook 4430s	HP	4096MB	Intel(R) Core(TM) i5-2450M CPU @ 2.50GHz (4 CPUs), ~2.5GHz	CNU207266X	1883	ECUIOWL01883	Windows 7 Professional 64-bit (6.1, Build 7601)
Carlos Mendoza	2100493564	Laptop	EliteBook 8460p	HP	8192MB	Intel(R) Core(TM) i7-3520M CPU @ 2.90GHz (4 CPUs), ~2.9GHz	S/N	1869	ECUIOWL01868	Windows 7 Professional 64-bit (6.1, Build 7601)
MARKETING										
Cristina Pérez	1.7059E+10	Laptop	PORTEGE Z930	Toshiba	6144MB	Intel(R) Core(TM) i5-3317U CPU @ 1.70GHz (4 CPUs), ~1.7GHz	3D052240H	2074364	ECUIOWL01651	Windows 7 Professional 64-bit (6.1, Build 7601)
Erika Muñoz	1723035331	Laptop	ProBook 4440s	HP	4096MB	Intel(R) Core(TM) i5-3210M CPU @ 2.50GHz (4 CPUs), ~2.5GHz	2CE2361B9Q	2073602	ECUIOWL2073602	Windows 7 Professional 64-bit (6.1, Build 7601)

Sebastián Bailón	1718948985	PC	Compaq 6200 Pro SFF PC	HP	2048MB	Intel(R) Core(TM) i5-2400 CPU @ 3.10GHz (4 CPUs), ~3.1GHz	MXL19080XZ	2073633	ECUIOWD0096	Windows 7 Professional 32-bit (6.1, Build 7601)
ADMINISTRATIVOS										
Jenny Sevillano	1721774147	PC	Compaq Pro 6300 SFF	HP	4096MB	Intel(R) Core(TM) i5-3470 CPU @ 3.20GHz (4 CPUs), ~3.2GHz	MXL3021SSM	2044244	ECUIOWD01936	Windows 8 Pro 64-bit (6.2, Build 9200)
Pasante Archivo	_____	PC	DH55PJ	Clon	8192MB	Intel(R) Core(TM) i7 CPU 870 @ 2.93GHz (8 CPUs), ~2.9GHz	S/N	2038281	ECUIOWD00355	Windows 8 Pro 64-bit (6.2, Build 9200)
Servicios Generales	_____	PC	Compaq dc5700 Microtower	HP	1024MB	Intel(R) Core(TM)2 CPU 6400 @ 2.13GHz (2 CPUs), ~2.1GHz	MXJ73503CT	2038977	ECUIOWD00539	Windows 7 Professional 64-bit (6.1, Build 7601)
Recepción	1713037941	PC	OptiPlex 7010	Dell	4096MB	Intel(R) Core(TM) i5-3470 CPU @ 3.20GHz (4 CPUs), ~3.2GHz	3KL5RW1	2047948	ECUIOWD001670	Windows 7 Professional 64-bit (6.1, Build 7601)
TICS										

José Navas	603885104	Laptop	EliteBook 8470p	HP	8192MB	Intel(R) Core(TM) i7- 3520M CPU @ 2.90GHz (4 CPUs), ~2.9GHz	CNU329C19P	2051105	ECUIOWL20511 05	Windows 8.1 Enterprise 64-bit (6.3, Build 9600)
José Santacruz	180344792 7	Laptop	Latitude E6440	Dell	4096MB	Intel(R) Core(TM) i7- 4600M CPU @ 2.90GHz (4 CPUs), ~2.9GHz	7XHVZ1	2069223	ECUIOWL20692 23	Windows 7 Professional 64-bit (6.1, Build 7601)
Luis Vallejo	171685748 5	Laptop	EliteBook 8460p	HP	4096MB	Intel(R) Core(TM) i7- 2620M CPU @ 2.70GHz (4 CPUs), ~2.7GHz	CNU20105B0	2051204	ECUIOWL20512 04	Windows 8.1 Enterprise 64-bit (6.3, Build 9600)
Paul Arellano	171333906 5	Laptop	SATELLITE U925t	Toshiba	4096MB	Intel(R) Core(TM) i5- 3337U CPU @ 1.80GHz (4 CPUs), ~1.8GHz	6D045520H	20140730	ECUIOWL00168 7	Windows 8.1 Enterprise 64-bit (6.3, Build 9600)
TALENTO HUMANO										
Cesar Montenegro	171586289 0	Laptop	PORTEGE Z930	Toshiba	6144MB	Intel(R) Core(TM) i5- 3317U CPU @ 1.70GHz (4 CPUs), ~1.7GHz	ZCO20350	2069322	ECUIOWL20693 22	Windows 7 Professional 64-bit (6.1, Build 7600)

Daniel Moya	170985128 9	Laptop	ProBook 4540s	HP	4096MB	Intel(R) Core(TM) i5- 3210M CPU @ 2.50GHz (4 CPUs), ~2.5GHz	2CE2402050	2074809	ECUIOWL02074	Windows 7 Professional 64-bit (6.1, Build 7601)
Eugenia Córdova	171211154 9	Laptop	Latitude E6440	Dell	4096MB	Intel(R) Core(TM) i7- 4600M CPU @ 2.90GHz (4 CPUs), ~2.9GHz	3WWRWZ1	2080235	ECOCCWL20802 35	Windows 7 Professional 64-bit (6.1, Build 7601)
Evelyn Lascano	172354295 5	Laptop	Latitude E6440	Dell	4096MB	Intel(R) Core(TM) i7- 4600M CPU @ 2.90GHz (4 CPUs), ~2.9GHz	1MXYXZ1	2080433	ECUIOWL20804 33	Windows 7 Professional 64-bit (6.1, Build 7601)
Elearning 1	_____	PC	Compaq dc5700 Microtower	HP	2040MB	Intel(R) Core(TM)2 CPU 6400 @ 2.13GHz (2 CPUs)	MXJ73808G5	2038137	ECUIOWL20381 37	Windows XP Professional (5.1, Build 2600)
Elearning 2	_____ -	PC	Compaq dc5700 Microtower	HP	2048MB	Intel(R) Core(TM)2 CPU 6400 @ 2.13GHz (2 CPUs), ~2.1GHz	MXJ80300N5	2074258	ECUIOWD00256	Windows 7 Professional 64-bit (6.1, Build 7601)

Estefanía Paredes	172458671 2	Laptop	ProBook 4430s	HP	4096MB	Intel(R) Core(TM) i3- 2310M CPU @ 2.10GHz (4 CPUs), ~2.1GHz	CNU1150KVW	2073831	ECUIOWL01481	Windows 7 Professional 64-bit (6.1, Build 7600)
Luis Arizaga	171772828 9	Laptop	EliteBook 2530p	HP	4096MB	Intel(R) Core(TM)2 Duo CPU L9400 @ 1.86GHz (2 CPUs), ~1.9GHz	CND0440054	2072155	ECUIOWL01850	Windows 7 Professional 64-bit (6.1, Build 7601)
José Carpio	171358003 1	Laptop	Latitude E6440	Dell	4096MB	Intel(R) Core(TM) i7- 4600M CPU @ 2.90GHz (4 CPUs), ~2.9GHz	25RRWZ1	2080174	ECUIOWL20801 74	Windows 7 Professional 64-bit (6.1, Build 7601)
María Dueñas	172244386 6	Laptop	Latitude E6440	Dell	4096MB	Intel(R) Core(TM) i7- 4600M CPU @ 2.90GHz (4 CPUs), ~2.9GHz	63YRWZ1	2080143	ECUIOWL20801 43	Windows 7 Professional 64-bit (6.1, Build 7601)
Manuel Jiménez	172084878 5	Laptop	Latitude E6440	Dell	4096MB	Intel(R) Core(TM) i7- 4600M CPU @ 2.90GHz (4 CPUs), ~2.9GHz	GX3HVZ1	2069506	ECOCCWL20695 06	Windows 7 Professional 64-bit (6.1, Build 7601)

Oswaldo Peña	1710427350	Laptop	PORTEGE Z930	Toshiba	6144MB	Intel(R) Core(TM) i5-3317U CPU @ 1.70GHz (4 CPUs), ~1.7GHz	3D052195H	2072490	ECUIOWL207249	Windows 8 Pro 64-bit (6.2, Build 9200)
Patricia Mosquera	1718489337	Laptop	ProBook 4440s	HP	4096MB	Intel(R) Core(TM) i5-3210M CPU @ 2.50GHz (4 CPUs), ~2.5GHz	2CE2361BDF	2030049	ECUIOWL01879	Windows 7 Professional 64-bit (6.1, Build 7601)
Reserva Vuelos	— —	PC	Compaq Pro 6300 MT	HP	4096MB	Intel(R) Core(TM) i5-3470 CPU @ 3.20GHz (4 CPUs), ~3.2GHz	MXL24120NP	2072599	ECUIOWD01886	Windows 7 Professional 64-bit (6.1, Build 7601)
Reserva Vuelos 2	— —	PC	Compaq dc5700 Microtower	HP	2048MB	Intel(R) Core(TM)2 CPU 6400 @ 2.13GHz (2 CPUs), ~2.1GHz	MXJ7340BQ4	2072771	ECUIOWD01747	Windows 7 Professional 64-bit (6.1, Build 7600)
Sebastián Flores	1718513888	Laptop	Latitude E6440	Dell	4096MB	Intel(R) Core(TM) i7-4600M CPU @ 2.90GHz (4 CPUs), ~2.9GHz	S/N Borrada	2080457	ECUIOWL2080457	Windows 7 Professional 64-bit (6.1, Build 7601)

Silvia Campaña	171116701 3	Laptop	ProBook 4440s	HP	4096MB	Intel(R) Core(TM) i5- 3210M CPU @ 2.50GHz (4 CPUs), ~2.5GHz	2CE242YVQ	2072308	ECUIOWL01891	Windows 7 Professional 64-bit (6.1, Build 7601)
INVESTIGACION Y DESARROLLO										
Anabel Arauz	172107614 7	Laptop	ProBook 4440s	HP	4096MB	Intel(R) Core(TM) i5- 3210M CPU @ 2.50GHz (4 CPUs), ~2.5GHz	DL19QW1	2037079	ECUIOWL01854	Windows 7 Professional 64-bit (6.1, Build 7601)
Iveth Carchipulla	171663627 7	Laptop	Precision M4800	Dell	16384MB	Intel(R) Core(TM) i7- 4600M CPU @ 2.90GHz (4 CPUs), ~2.9GHz	CNU6734455	2080228	ECUIOWL20802 28	Windows 8 Pro 64-bit (6.2, Build 9200)
Jorge Soria	180379096 1	Laptop	ProBook 4440s	HP	4096MB	Intel(R) Core(TM) i5- 3210M CPU @ 2.50GHz (4 CPUs), ~2.5GHz	CNU34567234	2036003	ECUIOWL02071	Windows 7 Professional 64-bit (6.1, Build 7601)
Leonardo Mena	171488235 2	Laptop	EliteBook 8470p	HP	4096MB	Intel(R) Core(TM) i7- 3520M CPU @ 2.90GHz (4 CPUs), ~2.9GHz	CNU319NKYN	1677	ECUIOWL01677	Windows 7 Professional 64-bit (6.1, Build 7601)

Jean Gonzalez	1719501684	Laptop	Satélite p55t	Toshiba	16384MB	Intel(R) Core(TM) i7-3520M CPU @ 2.90GHz (4 CPUs), ~2.9GHz	7D012446M	2069667	ECUIOWL2069667	Windows 8 Pro 64-bit (6.2, Build 9200)
Augusto Yépez	1715853410	Laptop	Satélite p55t	Toshiba	16384MB	Intel(R) Core(TM) i7-3520M CPU @ 2.90GHz (4 CPUs), ~2.9GHz	7D012522M	2069674	ECUIOWL2069674	Windows 8 Pro 64-bit (6.2, Build 9200)
Wilson Espinoza	602433468	Laptop	Portege Z930	Toshiba	6144MB	Intel(R) Core(TM) i5-3210M CPU @ 2.50GHz (4 CPUs), ~2.5GHz	2D037462H	205103	ECUIOWL205103	Windows 7 Professional 64-bit (6.1, Build 7601)
GERENTES										
Margarita Rodríguez	1707235295	Laptop	ULTRA BOOK	Toshiba	6144MB	Intel(R) Core(TM) i5-3317U CPU @ 1.70GHz (4 CPUs), ~1.7GHz	2B034671H	2038823	ECUIOWL2038823	Windows 7 Professional 64-bit (6.1, Build 7601)
Edwin Moreno	1706892229	Laptop	ULTRA BOOK PT234P	Toshiba	6144MB	Intel(R) Core(TM) i5-3317U CPU @ 1.70GHz (4 CPUs), ~1.7GHz	3D052269H	2038540	ECUIOWL2038540	Windows 7 Professional 64-bit (6.1, Build 7601)

Juan Fernando Cabrera	1712751138	Laptop	ULTRA BOOK PA5013U	Toshiba	6144MB	Intel(R) Core(TM) i5-3317U CPU @ 1.70GHz (4 CPUs), ~1.7GHz	3D052258H	2069964	ECUIOWL2069964	Windows 7 Professional 64-bit (6.1, Build 7601)
Elena Sánchez	1719053108	Laptop	Latitude E6440	Dell	4096MB	Intel(R) Core(TM) i7-3520M CPU @ 2.90GHz (4 CPUs), ~2.9GHz	66PTD12	2080501	ECUIOWL2080501	Windows 7 Professional 64-bit (6.1, Build 7601)
Paulo López	1802259687	Laptop	PROBOOK430S	HP	4096MB	Intel(R) Core(TM) i5-3317U CPU @ 1.70GHz (4 CPUs), ~1.7GHz	CNU2072698	2070106	ECUIOWL2070106	Windows 7 Professional 64-bit (6.1, Build 7601)
Bernardo Traversari	1705626867	Laptop	Portege Z930	Toshiba	6144MB	Intel(R) Core(TM) i5-3317U CPU @ 1.70GHz (4 CPUs), ~1.7GHz	3D052120H	2069735	ECUIOWL2069735	Windows 8.1 Enterprise 64-bit (6.1, Build 7601)
Mauricio Aguinaga	1711845402	Laptop	Portege Z930	Toshiba	6144MB	Intel(R) Core(TM) i5-3317U CPU @ 1.70GHz (4 CPUs), ~1.7GHz	3D052108H	2071066	ECUIOWL2071066	Windows 7 Professional 64-bit (6.1, Build 7601)

Ricardo Dávalos	1709626533	Laptop	Portege Z930	Toshiba	6144MB	Intel(R) Core(TM) i5-3317U CPU @ 1.70GHz (4 CPUs), ~1.7GHz	3D052242H	2069742	ECUIOWL2069742	Windows 7 Professional 64-bit (6.1, Build 7601)
Adrián Lovato	1709213514	Laptop	ELITE BOOK 8470P	HP	4096MB	Intel(R) Core(TM) i7-3520M CPU @ 2.90GHz (4 CPUs), ~2.9GHz	CNU3499KNN	2068639	ECUIOWL2068639	Windows 7 Professional 64-bit (6.1, Build 7601)
Laura Álvarez	1711992568	Laptop	PAVILLON DV4	HP	4096MB	Intel(R) Core(TM) i5-3317U CPU @ 1.70GHz (4 CPUs), ~1.7GHz	CNU12427WQ	2068714	ECUIOWL2068714	Windows 7 Professional 64-bit (6.1, Build 7601)
Manuel Muñoz	6101182795	Laptop	Portege Z930	Toshiba	6144MB	Intel(R) Core(TM) i5-3317U CPU @ 1.70GHz (4 CPUs), ~1.7GHz	3E024482H	2069438	ECUIOWL2069438	Windows 7 Professional 64-bit (6.1, Build 7601)
PLANTA ASME										
Cristian iza	1714960174	Pc	Hp compaq 6200 pro mt pc	Hp compaq	4gb	Intel64 family 6 model 42 stepping 7 genuineintel 3101mhz	Mxl2022h4t	2000063894	Ecoccwd01588	Microsoft windows 7 professional 6.1.7601 sp1 compilacion 7601
		Pc			4gb		Mxl203045h	2000063948	Ecoccwd01590	

Juan montalvan	2100420963		Hp compaq 6200 pro mt pc	Hp compaq		Intel64 family 6 model 42 stepping 7 genuineintel 1581mhz				Microsoft windows 7 professional 6.1.7601 sp1 compilacion 7601
Narcisa cedeño	2200031728	Laptop	Dell system xps i502x	Dell	4gb	Intel64 family 6 model 42 stepping 7 geniuneintel 2001 mhz	Fp0zpq1	2000045456	Ecuow101531	Microsoft windows 7 professional 6.1.7601 sp1 compilacion 7601
Byron barahona	502970155	Laptop	Hp probook 4430s	Hp	4gb	Intel64 family 6 model 42 stepping 7 genuineintel 2501mhz	Cnu2072632	2000040765	Ecocw102094	Microsoft windows 7 professional 6.1.7601 sp1 compilacion 7601
Bodega despachos										
Ciro villamarin	603444605	Pc	Hp compaq dc5700 microtower	Hp compaq	2gb	X64 family 6 model 15 stepping 2 geniuneintel 2133mhz	Mxj7340bqv	2000026455	Ecocwd00698	Microsoft windows 7 professional 6.1.7600 sp1 compilacion 7600
Danny leon	2100262688	Pc	Hp compaq dx2300 microtower	Hp compaq	512mb	X64 family 6 model 15 stepping 13 geniuneintel 1600 mhz	Mxl8020xdd	2000026394	Ecocwd01121	Microsoft windows 7 professional 6.1.7600 sp1 compilacion 7600
Leonardo izurieta	1500752827	Pc	Hp compaq pro 6300 mt	Hp compaq	4gb	Intel64 family 6 model 58 stepping 9 genuineintel 3201mhz	mxl24120nv	2000026356	Ecocwd01802	Microsoft windows 7 professional 6.1.7601 sp1 compilacion 7601

Superintendencia de proyectos										
Superintendente facilidades- danny frias	180333951 2	Pc	H81mlc	Biostar	8gb	Intel(r) core(tm) i7-4770 cpu @ 3.40ghz	None	2000028602	Ecocccwd002355	Microsoft windows 7 ultimate 6.1.7601 sp1 compilacion 7601
Planta de producción										
Andres guevara	603790890	Pc	Hp z200 workstation	Hp	8gb	Intel(r) xeon(r) x3430 @ 2.40ghz	2ua0291w0w	2000033590	Ecocccwd001424	Microsoft windows 7 professional 6.1.7601 sp1 compilacion 7601
Gonzalo villamarin	180277449 5	Pc	Hp compaq elite 8300 cmt	Hp compaq	4gb	Intel(r) core(tm) i7-3770 cpu @ 3.4ghz	Mxl2490j1r	2000030537	Ecocccwd01760	Microsoft windows 7 professional 6.1.7601 sp1 compilacion 7601
Galo zurita	601791924	Pc	Hp compaq 6000 pro mt	Hp compaq	2gb	Intel core 2 quad cpu q8400 @ 2.66ghz	Mxl1070jzx	2e+10	Ecocccwd01840	Microsoft windows 7 professional 6.1.7600 sp1 compilacion 7600
Mario gualotuna		Laptop	Hp probobook 4430s	Hp	4gb	Intel core i5- 2450m cpu @ 2.5ghz	Cnu213321g	2000030292	Ecocccwl01629	Microsoft windows 7 professional 6.1.7601 sp1 compilacion 7601
Torno		Pc	Hp pavilion p6000	Hp	8gb		4cw019183t	No tiene	Sertecpet-pc	Microsoft windows 7

						Intel core 2 quad q8300 @ 2.5ghz				professional 6.1.7601 sp1 compilacion 7601
Proyectos										
Asistente adm proyectos		Pc	Optiplex 7010	Dell	4gb	Intel core i5- 3470 cpu @ 3.2ghz	4tnlrw1	2000067113	Ecoccwd001606	Microsoft windows 7 professional 6.1.7601 sp1 compilacion 7601
Janina quinde	110461341 7	Pc	Hp compaq elite 8300 cmt	Hp compaq	4gb	Intel core i7- 3770 cpu @ 3.4ghz	Mxl2490g9j	2000063290	Ecoccwd01757	Microsoft windows 7 professional 6.1.7601 sp1 compilacion 7601
Luis achance	602914061	Laptop	Portege z930	Toshiba	6gb	Intel core i5- 3371u cpu @ 1.7ghz	3d052142h	2022	Ecoccwl002022	Microsoft windows 7 professional 6.1.7601 sp1 compilacion 7601
Rafael caiza	180317735 9	Laptop	Latitude e6440	Dell	4gb	Intel core i7- 4600m cpu @ 2.9ghz	J1tyxz1	2000080389	Ecoccwl2080389	Microsoft windows 8 enterprise 6.3.9600 n/d compilacion 9600
Bodega proyectos										

Bodega proyectos 1		Pc	Hp compaq 6000 pro mt	Hp compaq	4gb	Intel core 2 quad cpu q8400 2.67ghz	Mxl1070k0n	2000043919	Ecoccwd01813	Microsoft windows 7 professional 6.1.7601 sp1 compilacion 7601
Bodega proyectos 2		Pc	Optiplex 7010	Dell	4gb	Intel core i5-3470 cpu 3.2ghz	4tljrw1	2000043940	Ecoccwd001667	Microsoft windows 7 professional 6.1.7601 sp1 compilacion 7601
Administración										
Angel tandazo	2100077508	Pc	Hp compaq pro 6300 sff	Hp compaq	4gb	Intel core i5-3470 cpu 3.2ghz	Mxl2360wmc	2000065829	Ecoccwd01823	Microsoft windows 7 professional 6.1.7601 sp1 compilacion 7601
Freddy nuñez	1600175572	Pc	Hp compaq pro 6300 sff	Hp compaq	4gb	Intel core i5-3470 cpu 3.2ghz	Mxl2360fc9	Mxl2360fc9	Ecoccwd01822	Microsoft windows 7 professional 6.1.7601 sp1 compilacion 7601

Juan avila	171483876 8	Pc	Hp compaq pro 6300 sff	Hp compaq	4gb	Intel core i5- 3470 cpu 3.2ghz	Mxl2360fbb	2000068202	Ecoccwd01812	Microsoft windows 7 professional 6.1.7601 sp1 compilacion 7601
Estefania vivanco	210078282 6	Pc	Optiplex 9020	Dell	4gb	Intel core i7- 4770 cpu 3.4ghz	1ksmk02	2000077518	Ecoccwd2077518	Microsoft windows 7 professional 6.1.7601 sp1 compilacion 7601
Nelson tandazo	171465991 7	Pc	Hp compaq 8200 elite sff	Hp compaq	8gb	Intel core i7- 2600 cpu 3.4ghz	Mxl1310mfw	2000068103	Ecuiowd01507	Microsoft windows 7 professional 6.1.7601 sp1 compilacion 7601
Seguridad										
Braulio santos	180283003 2	Laptop	Hp probook 4430s	Hp	4gb	Intel core i3- 2310m cpu 2.1ghz	Cnu1150k7v	2000077662	Ecoccwl01487	Microsoft windows 7 professional 6.1.7601 sp1 compilacion 7601

Cristhian rivas	220004854 0	Pc	Optiplex 9020	Dell	4gb	Intel core i7- 4770 cpu 3.4ghz	9q3wg02	2000069629	Ecoccd2069629	Microsoft windows 7 professional 6.1.7601 sp1 compilacion 7601
Daniel bermeo	171197015 0	Laptop	Latitude e6440	Dell	8gb	Intel core i7- 4600m cpu 2.9ghz	F7ntd12	2000080549	Ecoccd2080549	Microsoft windows 7 professional 6.1.7601 sp1 compilacion 7601
Fernando paredes	180329466 7	Pc	Hp compaq dc5700 microtower	Hp compaq	3gb	Intel core 2 cpu 6400 2.13ghz	Mxj7340bqs	2000027759	Ecoccd02050	Microsoft windows 7 professional 6.1.7601 sp1 compilacion 7601
Italo haro		Laptop	Hp pavilion 14 notebook	Hp	4g	Intel core i7- 4702mq cpu 2.2ghz	5cd3342pdw	2000068912	Ecoccd2068912	Microsoft windows 8.1 enterprise 6.3.9600 n/d compilacion 9600

Juan araque	180329688 6	Laptop	Hp probook 4430s	Hp	4gb	Intel core i5- 2450m cpu 2.5ghz	Cnu2132z0y	2000041731	Ecoccwl01608	Microsoft windows 7 professional 6.1.7601 sp1 compilacion 7601
Medico sachá		Pc	Hp compaq dc5700 microtower	Hp	3gb	Intel core 2 cpu 6400 2.13ghz	Mxj73808h6	2000027155	Ecoccwd027155	Microsoft windows 8 pro 6.2.9200 n/d compilacion 9200
Medico coca		Pc	Optiplex 9020	Dell	4gb	Intel core i7- 4770 cpu 3.4ghz	85k7fz1	1729	Ecoccwd001729	Microsoft windows 8 pro 6.2.9200 n/d compilacion 9200
Santiago simbaña	171551352 7	Laptop	Hp probook 4430s	Hp	4gb	Intel core i5- 2450m cpu 2.5ghz	Cnu21331qj	2000042899	Ecoccwl2042899	Microsoft windows 7 professional 6.1.7601 sp1 compilacion 7601

Facturación

Angel valdiviezo	601911886	Pc	Optiplex 7010	Dell	4gb	Intel core i5- 3470 cpu 3.2ghz	4tllrw1	2000064839	Ecoccwd001611	Microsoft windows 7 professional 6.1.7601 sp1 compilacion 7601
Diana lapo	210063314 4	Pc	Hp compaq pro 6300 sff	Hp compaq	4gb	Intel core i5- 3470 cpu 3.2ghz	Mx12370wvh	2000067274	Ecoccwd01827	Microsoft windows 7 professional 6.1.7601 sp1 compilacion 7601
Diana prieto	171723345 4	Pc	Optiplex 7010	Dell	4gb	Intel core i5- 3470 cpu 3.2ghz	4v0krw1	2000066529	Ecoccwd001610	Microsoft windows 7 professional 6.1.7601 sp1 compilacion 7601
Joffre cortez	401313887	Laptop	Hp compaq 6710b	Hp compaq	2gb	Intel core 2 duo cpu t8300 2.4ghz	Cnu82824gp	1174	Ecoccwl01174	Microsoft windows 7 professional 6.1.7601 sp1 compilacion 7601

Monica hernandez	1721942819	Pc	Hp compaq pro 6300 sff	Hp compaq	4gb	Intel core i5-3470 cpu 3.2ghz	Mxl2360wmm	2000064822	Ecoccwd01826	Microsoft windows 7 professional 6.1.7601 sp1 compilacion 7601
Sandy cherrez	1718327297	Pc	Hp compaq pro 6300 sff	Hp compaq	4gb	Intel core i5-3470 cpu 3.2ghz	Mxl2360wmy	2000067281	Ecoccwd01829	Microsoft windows 7 professional 6.1.7601 sp1 compilacion 7601
Supervisión producción-joffre cortez	401313887	Pc	Hp compaq 6200 pro mtc	Hp compaq	4gb	Intel core i5-2400 cpu 3.1ghz	Mxl1330lj2	2000064778	Ecoccwd01839	Microsoft windows 7 professional 6.1.7601 sp1 compilacion 7601
Talento humano coca										
Alex carrion	603572744	Laptop	Portege z930	Toshiba	6gb	Intel core i5-3371u cpu 1.7ghz	Zc018857h	2000027513	Ecoccwl0027513	Microsoft windows 8 pro 6.2.9200 n/d compilacion 9200

Ana caicedo	170850836 9	Laptop	Nc75xx	Hp	2gb	Intel core 2 duo t8300 2.4ghz	Cnu8131fw9	2000030001	Ecoccwl2030001	Microsoft windows 8 pro 6.3.9600 n/d compilacion 9600
Recepción- doris Bustamante	220007794 5	Pc	Hp compaq 6200 pro mtc	Hp compaq	2gb	Intel core i5- 2400 cpu 3.1ghz	Mx113808dw	2000028367	Ecoccwd01561	Microsoft windows 7 professional 6.1.7601 sp1 compilacion 7601
Katy Hinojosa	172439987 6	Laptop	Latitude e6440	Dell	4gb	Intel core i7- 4600m cpu 2.9ghz	4ggrwz1	2000080214	Ecoccwd2080214	Microsoft windows 7 professional 6.1.7601 sp1 compilacion 7601
Jimmy anchatuña	172133580 8	Laptop	Latitude e6440	Dell	4gb	Intel core i7- 4600m cpu 2.9ghz	I3qrwz1	2000080044	Ecoccwd2080044	Microsoft windows 7 professional 6.1.7601 sp1 compilacion 7601
Tics coca										

Santiago molina	171344961 7	Laptop	Latitude e6440	Dell	8gb	Intel core i7- 4600m cpu 2.9ghz	Bbgmyz1	2000080556	Ecoccw12080556	Microsoft windows 8.1 enterprise 6.3.9600 n/d compilacion 9600
Javier Ledesma	172365524 5	Laptop	Latitude e6440	Dell	8gb	Intel core i7- 4600m cpu 2.9ghz	8dwyxz1	2000080365	Ecoccw12080365	Microsoft windows 8.1 enterprise 6.3.9600 n/d compilacion 9600
Jose choca	131135879 8	Laptop	Latitude e6440	Dell	4gb	Intel core i7- 4600m cpu 2.9ghz	Gb6myz1	2000080563	Ecooccw12080563	Microsoft windows 8.1 enterprise 6.3.9600 n/d compilacion 9600
Gerencia coca										
Hernán velastegui	601739279	Laptop	Hp elitebook 8470p	Hp	4gb	Intel core i7- 3540m cpu 3ghz	Cnu329c199	2166	Ecoccw1002166	Microsoft windows 7 professional 6.1.7601 sp1 compilacion 7601

Garita base coca										
Garita		Pc	Hp compaq 6000 pro mt	Hp compaq	2gb	Intel(r) core(tm) 2 quad cpu q8400 @ 2.66ghz	Mx102710cq	2000029173	Ecoccwd2029173	Microsoft windows 7 professional 6.1.7601 sp1 compilacion 7601
Facilidades										
Wilmer vaca	503187825	Laptop	Latitude e6440	Dell	4gb	Intel core i7-4600m cou @ 2.9ghz	8k3vd12	2000080518	Ecoccwl2080518	Microsoft windows 7 professional 6.1.7601 sp1 compilacion 7601
Diego Pérez	210032385 2	Laptop	Latitude e6440	Dell	4gb	Intel core i7-4600m cou @ 2.9ghz	9zmttd12	2000080488	Ecoccwl2080488	Microsoft windows 7 professional 6.1.7601 sp1 compilacion 7601

Patricio delgado	401174990	Laptop	Latitude e6440	Dell	4gb	Intel core i7-4600m cou @ 2.9ghz	7hrt12	2000080464	Ecocw12080464	Microsoft windows 7 professional 6.1.7601 sp1 compilacion 7601
Hugo Alvares	502780117	Laptop	Latitude e6440	Dell	4gb	Intel core i7-4600m cou @ 2.9ghz	2lwyx1	2000080358	Ecocw12080358	Microsoft windows 7 professional 6.1.7601 sp1 compilacion 7601

Anexo 5. Guía de entrevista realizada al coordinador de activos fijos

¿Cómo ha percibido el control de activos luego de la implementación del sistema?

Después de haber realizado la implementación del sistema y puesta en marcha del mismo se puede constatar que efectivamente se posee un mayor control de los equipos electrónicos.

¿Cuál es el proceso que actualmente se maneja para el control de los activos informáticos?

Cada activo informático está asociado a una etiqueta RFID, la cual es registrada tanto a la entrada como a la salida para su control, la etiqueta es suministrada por el departamento de TIC'S.

¿Si existiera la pérdida de un activo, que se realiza?

Se procede a buscar en el sistema al responsable del activo, posteriormente se contacta a la persona y se le pregunta por el equipo, en el caso de que la persona no estuviese en la empresa, se contactara con talento humano para la ubicación del mismo, de ser el caso será descontado del rol o de su liquidación.

¿En su opinión tiene algún comentario al respecto del nuevo sistema?

El sistema está funcionando correctamente y nos ayudado a tener un reporte al día de los equipos, lo que ha reducido un costo para la empresa, como recomendación seria optimo que el sistema sea web para facilitar su apertura desde cualquier parte.

Anexo 6. Guía de entrevista realizada al coordinador de TI

¿Cuál es el proceso que actualmente se realiza cuando ingresa un activo informático al departamento de TI?

Si un activo informático entra al departamento de TI, en primer lugar hay que prepararlo, es decir instalando los programas base que se utiliza dentro de la empresa, si se requiere algún otro programa para su instalación se lo realiza mediante el pedido del departamento al que se le va a asignar dicho equipo, se coloca su etiqueta de RFID y se ingresan los datos del equipo en el sistema, para realizar la descarga del equipo fuera del departamento de TI se necesita un acta de entrega esta se genera de manera automática en el sistema de control de activos informáticos, una vez generada el acta es firmada por el Jefe de TI.

¿Cree usted que se lleva un control efectivo de los activos informáticos?

Actualmente con el beneficio del sistema de control de activos podemos llevar un control de los equipos de manera actualizada y periódica, hoy en día todos los equipos son obligados a pasas por el departamento de TIC para ser colocados su etiqueta respectiva, este proceso se lo realiza tanto en Quito como en el Coca.

¿Cómo ha apreciado la tecnología RFID?

Es una tecnología muy interesante ha facilitado de manera considerable las necesidades que teníamos como departamento al evitar la pérdida de equipos que teníamos, se está estudiando el proyecto para ser ubicado en otros proyectos de la empresa.

¿En su opinión tiene algún comentario al respecto del nuevo sistema?

Se necesita que el sistema migre a web lo que permita acceder al mismo desde cualquier ordenador perteneciente a la empresa, de igual forma se está planificando la puesta en marcha del nuevo proyecto.

Anexo 7. Guía de entrevista realizada al supervisor de seguridad

¿Existe alguna política sobre el uso de equipos fuera de la empresa?

Actualmente la empresa nos proporcionó una política la cual nos indica que ninguna persona puede abandonar la empresa si no es registrada en el caso que posea maleta o mochila, si llevase un equipo informático este debe ser registrado en el sistema.

¿Cómo lleva el control de equipos el personal de seguridad?

Los equipos registrados en el sistema según el empleado este ingresando o saliendo de la empresa con un equipo, el sistema permite que el guardia de turno elija entre estas 2 opciones una vez ubicado el código de RFID se cargan en la pantalla la información del equipo y de la persona responsable.

¿Qué sucede con esta información?

Actualmente la información como en su momento nos supieron manifestar es almacenada en una base de datos, la cual permite emitir reportes y consultas de manera inmediata.

¿Si existiera la pérdida de un activo, que se realiza?

Se procedería a realizar una búsqueda en el sistema por su N° de Activo o responsable del equipo, previo aviso al departamento de TIC'S y Activos Fijos,

¿En su opinión tiene algún comentario al respecto del nuevo sistema?

Solo se puede recalcar que nos ha brindado una gran ayuda debido a que somos una empresa contratada la cual no tenía un respaldo por parte de Sertecpet para validar una política de entrada y salida de equipos por lo que la culpa recaía sobre nosotros al momento de que un equipo se extravié.

Anexo 8. Factura de consultora para la realización de los informes de activos

1792194512001

QUITO, 14 DE MAYO DEL 2014

SERTECPET S.A

TELÉFONO: 3954900

RUC: 179126330800

Guía de Remisión:

CANTIDAD	DESCRIPCION	V. UNITARIO	V. TOTAL
1	AVALUO (LIQUIDACION PROYECTO INSPECCION FISICA, CODIFICACION, CONCILIACION SEGUN NIFF)	\$ 15.977,49	\$ 15.977,49

SERTECPET
Diana Hinojosa
C.C. 1708063218
Control de Gestión - Proyectos

Andrés Enriquez Milton Manuel • CDL IDEA • RUC 1101985503001 • Aut. 1460 • 2401 334

erado del 3401 al 3600 • Fecha de caducidad: 23/Abril/2015

mal: Usuario - 1ra. Copia Rosada: Vendedor - 2da. Copia Amarilla: Sin Valor Tributario

SUBTOTAL	\$ 15.977,49
DESCUENTO	---
I.V.A. 0 %	\$ 0,00
I.V.A. 12%	\$ 1.917,30
TOTAL	\$ 17.894,79

FIRMA AUTORIZADA

RECIBIDO POR

Cabe recalcar que el costo para el levantamiento de equipos informáticos tiene un valor de \$5000 en la imagen se puede apreciar el costo total por los servicios ofrecidos por la empresa encargada.